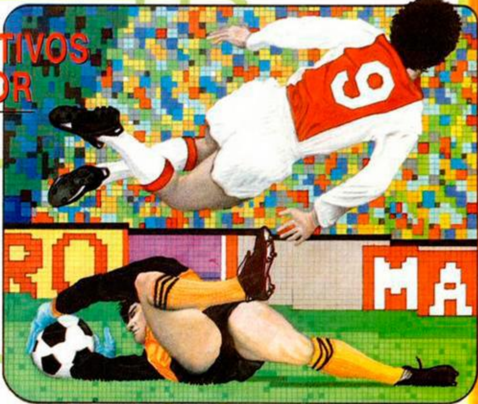
N.º 7
Ptas. 395
Canarias, Ceuta y Melilla, 375 ptas.
ESPECIAL

MICRO HOBBY

REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES SINCLAIR Y COMPATIBLES

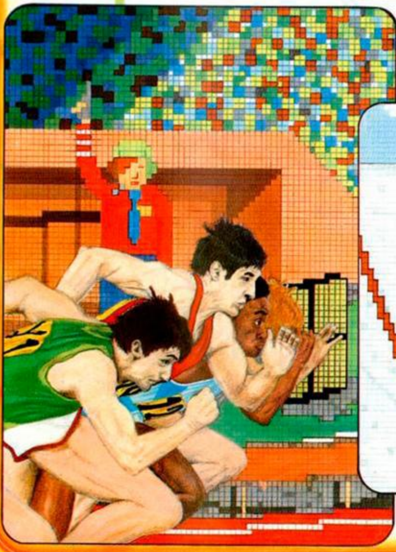
GUÍA CON TODOS LOS PROGRAMAS DEPORTIVOS PARA TU ORDENADOR



EDITOR DE PANTALLAS

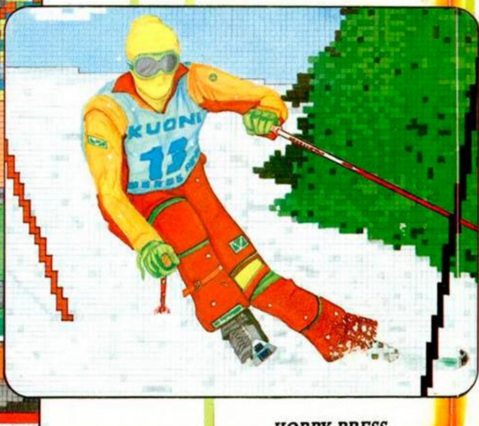
UN PROGRAMA-MONITOR QUE TE PERMITE CREAR MAPEADOS EN TRES DIMENSIONES

LAS VEINTE MEJORES RUTINAS DE CÓDIGO MÁQUINA PARA TU SPECTRUM



DISEÑADORES GRÁFICOS

UNA PROFESIÓN EN ALZA



HOBBY PRESS

MICRO HOBBY ESPECIAL

ESPECIAL MICROHOBBY•AÑO III•N.º 7 DICIEMBRE 1987



SUMARIO

Director Editorial
José I. Gómez-Centurión

Director
Gabriel Nieto

Director de Microhobby
Domingo Gómez

Maquetación
Berta Fernández

Redacción
Pedro Pérez, Cristina Fernández

Colaboradores
José J. García Quesada
Pablo Ariza

Secretaria Redacción
Carmen Santamaría

Jefe de Publicidad
Mar Lumbreras

Fotografía
Carlos Candel
Miguel Lamana

Dibujos
Luis Muñoz
Manuel Barco
Javier Igual

Edita
HOBBY PRESS, S.A.

Presidente
María Andrino

Consejero Delegado
José I. Gómez-Centurión

Subdirector General
Andrés Aylagas

Director Gerente
Fernando Gómez-Centurión

Jefe de Administración
José Ángel Jiménez

Jefe de Producción
Carlos Peropadre

Jefe de Marketing
Javier Bermejo

Suscripciones
Tel. 734 65 00

Redacción, Administración y Publicidad
Ctra. de Irún km 12,400
28049 Madrid
Tel. 734 70 12. Telefax 734 82 98

Dto. Circulación
Paulino Blanco

Distribución
Coedis, S. A. Valencia, 245
Barcelona

Imprime
LERNER

Fotocomposición
Novocomp, S. A.
Nicolás Morales, 38-40

Fotomecánica
Ibérico

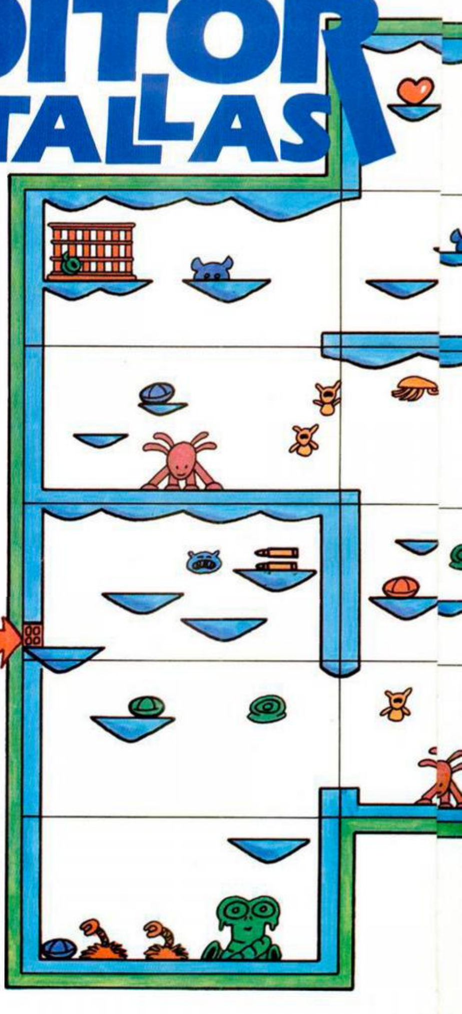
Depósito legal: M-36.598-1984

Representante para Argentina, Chile, Uruguay y Paraguay, Cía. Americana de ediciones, S.R.L. Sud América 1.532. Tel.: 21 24 64. 1290 BUENOS AIRES (Argentina).

Solicitado control
OJD

EDITOR DE PANTALLAS

En el último especial os hablamos de uno de los métodos más comunes para realizar el mapeado de las pantallas de un juego; pues bien, con este editor os resultará muy fácil construir pantallas a partir de un conjunto de gráficos e incluirlas en vuestros propios juegos, y todo ello sin tener que trabajar en ensamblador en ningún momento.



Quizá los poco más de 2.800 bytes del código máquina y las 420 líneas del programa Basic no resulten demasiado alentadores, pero si te decides a leer el siguiente artículo es posible que el trabajo de teclearlos en tu ordenador merezca la pena. Eso es, al menos, lo que nosotros esperamos.

PUESTA EN MARCHA

Una vez tecleado el basic del listado 1 y grabado a continuación el código del listado 2, puedes comenzar a trabajar con el editor.

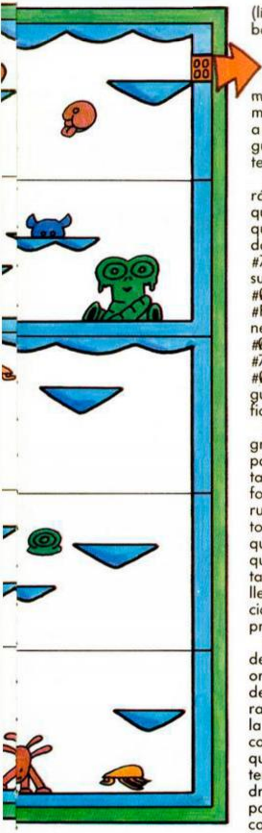
La primera decisión que deberás tomar será si quieres cargar los datos grabados en una sesión de trabajo anterior o almacenar los gráficos directamente de una pantalla. Si eliges la primera opción, se te pedirá el nombre del fichero que contiene esos datos (pulsa sólo ENTER para cargar el primero de la cinta) y cuando finalice la carga aparecerás directamente en el menú principal. Si, por el contrario, quieres introducir en memoria los gráficos que compondrán los decorados de las habitaciones tendrás que teclear el nombre del fichero de pantalla en él que los habrás almacenado previamente. Veámos cómo ha de hacerse esto.

LOS GRÁFICOS

El sistema elegido para la presentación de gráficos es el denominado por máscara, al que deben su éxito programas tales como Knight Lore o Alien 8, y que, sin duda, es sobradamente conocido por todos vosotros. En lo referente a los gráficos es el procedimiento que más memoria consume por necesitar, además del gráfico propiamente dicho, la máscara que le indica al ordenador (con los bits de la máscara activos) qué zonas del gráfico dejarán ver el fondo.

Normalmente el gráfico y la máscara suelen estar separados en memoria y almacenados en forma de scanes

J. J. García Quesada



(líneas de gráfico). Sin embargo, en nuestro caso la disposición es diferente; por motivos prácticos (la rutina de impresión es más sencilla y, por tanto, más corta) y es la siguiente: a cada scan del gráfico le sigue el scan correspondiente de la máscara.

Con un ejemplo lo verás rápidamente. Imaginemos que tenemos un gráfico (pequeño) compuesto por los datos : # 00, #7C, #70, #70, #48, #44, #02, #00 y su máscara por: #01, #01, #01, #03, #01, #00, #10, #F8. Nuestro programa los necesita almacenados así: #00-#01, #7C-#01, #70-#01, #70-#03, #48-#01, #44-#00, #02-#10 y #00-#F8 (los guiones separan scanes gráfico-máscara).

Si ya tienes creados tus gráficos con la máscara separada, el programa del listado 3 los convertirá a un formato manejable por la rutina de impresión del editor. El único problema es que en ese caso serás tu quien deberás construir la tabla de gráficos (ver detalles técnicos para ampliación), a menos que utilices el programa del listado 4.

En él, cada línea de datos deberá contener por este orden: la dirección de inicio del gráfico, la de la máscara, las dimensiones X e Y y la posición de pantalla (en coordenadas PRINT) donde quieres ponerlo. Cuando termine de ejecutarse tendrás una pantalla formada por un conjunto de gráficos con su máscara inmediatamente a la derecha. Grábala en cinta para cargarla posteriormente desde el programa.

Si aún no los tienes almacenados (o si dispones de los bancos de UDGs originales) puedes utilizar cualquiera de los diseñadores gráficos que se comercializan para formar una pantalla que cumpla estas condiciones.

ALMACENAMIENTO DE LOS GRÁFICOS

Una vez tengas grabada la pantalla (o pantallas) con los gráficos puedes entrar en la opción correspondiente del editor. Tras indicar al ordenador el número de gráficos que quieres crear (255 como máximo y este valor no se puede variar a menos que quieras empezar de nuevo), tienes que dar al programa el nombre del fichero de pantalla que quieres cargar (ENTER carga el primero). A continuación entrarás en un pequeño menú con tres opciones:

Cargar pantalla: carga otra pantalla más para guardar en memoria sus gráficos.

Editar gráficos: un instante después de pulsar la "E" te encontrarás con la última pantalla cargada y una ventana (formada por líneas DRAW con OVER 1) en la esquina superior izquierda. Puedes moverla con O, P, Q y A, y ampliarla/reducirla con 5, 6, 7 y 8. Cuando pulses ENTER todo lo contenido dentro de la ventana, incluido lo que hay bajo las líneas que constituyen el marco, se almacenará en memoria. Asegurate que el marco abarca exactamente el gráfico, máscara incluida, que quieres pasar a memoria porque no existe posibilidad directa de rectificación después de dar al ENTER. Pulsa "F" para volver al menú.

Ir al menú principal: da por acabada esta fase y te da la posibilidad de utilizar un joystick con interface Kempston. Por último te



pregunta el formato de pantalla a usar, el número de pantallas que vas a crear (máximo 255 y tampoco se podrá cambiar más adelante) y salta al menú principal.

Dispones, además, de alguna información adicional: el número de gráficos almacenados y la cantidad de memoria disponible.

MENÚ PRINCIPAL

En el que existen las siguientes opciones:

"D" Diseñar pantalla: para crear una nueva pantalla, que es añadida al final de las ya existentes.

"S" Salvar datos: guarda en cinta los datos de los gráficos que hay en memoria, los de las pantallas y todas las tablas y variables necesarias para el correcto funcionamiento del programa.



"C" Cargar datos: recupera los datos grabados con la opción salvar datos.

"G" Generar código: genera el código adecuado para poder imprimir las pantallas que has creado sin tener que utilizar el editor. El programa te informa de la zona de memoria que ocupará y te permite salvarlo para un uso posterior. Si al seleccionar esta opción estabas trabajando con formato ventana deberás incluir en tus cálculos la zona destinada al buffer de pantalla (en modo pantalla completa se trabaja directamente sobre la pantalla normal). Lee los detalles técnicos para utilizar el código generado.

"V" Ver pantallas: te permite ver las pantallas (si las hubiese) creadas hasta el momento. Las opciones de que dispones aquí son:

"A" Alterar pantalla: para realizar cambios en la pantalla que se muestra en ese instante.

"C" Copiar pantalla: copia la pantalla actual al final de todas las existentes. La copia pasa a ser la pantalla seleccionada.

"B " Borrar pantalla: borra la pantalla en curso. Se pide confirmación antes de llevarlo a cabo.

"5,8" Ver otras pantallas. **"ENTER"**: vuelve al menú principal seleccionando la pantalla actual.

En todo momento se indica la pantalla activa.

"F" Formato de pantalla: con esta opción puedes cambiar el formato de la pantalla. Existen dos modos: pantalla completa o ventana. El modo ventana sólo te muestra la zona de la pantalla global que abarca esa ventana. Necesitas dar las coordenadas de su esquina superior izquierda y las dimensiones. Este modo sólo es operativo cuando estás en la opción «Ver pantallas» del menú principal.

LISTADO 1 EDITOR DE PANTALLAS

```
   1 REM ********************
         *                  *
         *   EDITOR DE      *
         *   PANTALLAS.     *
         *                  *
         *   POR J.J.G.Q.   *
         *                  *
         ********************
  10 CLEAR 62599: GO TO 930
  20
  30 REM ** MENU PRINCIPAL **
  40
  50 GO SUB 1650
  60 GO SUB KEY
  65 IF K$="F" THEN GO SUB 223:
GO TO 50
  70 IF K$="D" THEN GO TO 530
  80 IF K$="S" THEN GO TO 630
  90 IF K$="C" THEN GO TO 710
 100 IF K$="G" THEN GO TO 780
 110 IF K$<>"V" THEN GO TO 60
 120
 130 REM ** VER/SELECCIONAR **
 140
 150 CLS : IF PEEK NUMSCR=Z THEN
 GO SUB 1980: GO TO 50
 155 IF PEEK SCRED=Z THEN POKE S
CRED,PEEK NUMSCR
 160 RANDOMIZE USR PANTALLA
 162 PRINT #U;AT U,Z; INVERSE U;
"A"; INVERSE Z;"lterar "; INVERS
E U;"C"; INVERSE Z;"opiar "; INV
ERSE U;"B"; INVERSE Z;"orrar ";
INVERSE 1;"5"; INVERSE Z;","; IN
VERSE U;"8"; INVERSE Z;" P:";PE
EK SCRED,
 170 GO SUB KEY
 180 IF K$=CHR$ 13 THEN BEEP .1,
20: GO TO 50
 185 IF K$="A" THEN GO TO 580
 190 IF K$="C" THEN GO TO 470
 200 IF K$="B" THEN GO TO 270
 210 IF K$="5" OR K$="6" THEN GO
 TO 370
 220 IF K$="7" OR K$="8" THEN GO
 TO 420
 221 GO TO 170
 222
 223 REM ** CAMBIAR FORMATO **
 224
 225 CLS : INPUT "PANTALLA COMPL
ETA                O VENTANA (C/V
) ? "; LINE A$: IF A$<>"C" AND A
$<>"V" THEN GO TO 225
 226 IF A$="C" THEN POKE MODO,Z:
 RETURN
 227 INPUT "ESQUINA SUPERIOR
               IZQUIERDA (X/Y) ";PX
;"/";PY: IF PX<U OR PX>31 OR PY<
U OR PY>23 THEN GO TO 227
 228 INPUT "DIMESIONES DE LA VEN
TANA          (DIMX/DIMY) ? ";DX;"
/";DY: IF DX<U OR DX>32 OR DY<U
OR DY>22 OR DX+PX>32 OR DY+PY>24
 THEN GO TO 227
 229 POKE POSVEN,PX: POKE POSVEN
+U,PY: POKE DIRVEN,DX: POKE DIRV
EN+U,DY: POKE MODO,U: RETURN
 267
 268 REM ** BORRAR PANTALLA **
 269
 270 PRINT AT 20,Z; INK 7; PAPER
 2;" PULSA ENTER PARA BORRAR Y O
TRA    TECLA PARA VOLVER AL MENU
."
 280 GO SUB KEY
 290 IF K$<>CHR$ 13 THEN GO TO 1
60
 300 RANDOMIZE USR DELSCR
 310 IF NOT PEEK NUMSCR THEN GO
TO 150
 320 IF PEEK SCRED>PEEK NUMSCR T
HEN POKE SCRED,PEEK NUMSCR
 330 CLS : GO TO 160
 340
 350 REM * VER SIGUIENTE PANT. *
 360
 370 IF PEEK SCRED=U THEN BEEP .
1,20: GO TO 170
 380 POKE SCRED,PEEK SCRED-U: GO
 TO 160
 390
 400 REM * VER PANT. ANTERIOR *
 410
 420 IF PEEK SCRED=PEEK NUMSCR T
HEN BEEP .3,-10: GO TO 170
 430 POKE SCRED,PEEK SCRED+U: GO
 TO 160
 440
 450 REM * COPIA PANT. ACTUAL *
 460
 470 IF PEEK NUMSCR=MXPANT THEN
GO SUB 1990: GO TO 160
 480 RANDOMIZE USR COPY
 490 POKE SCRED,PEEK NUMSCR: GO
TO 160
 500
 510 REM ** DISENAR PANTALLA **
 520
 530 IF PEEK NUMSCR=MXPANT THEN
GO SUB 1990: GO TO 60
 540 RANDOMIZE USR MKSCR
 550
 560 REM ** ALTERAR PANTALLA **
 570
 580 IF (PEEK NUMSCR=Z) OR (PEEK
 SCRED=Z) THEN GO SUB 1970: GO T
O 60
 590 RANDOMIZE USR ALTSCR
 600
 610 REM ** SALVAR DATOS **
 620
 630 LET DAT=MAXBLO+18: FOR R=TA
BGRF TO TABGRF+16
 640 POKE DAT,PEEK R: LET DAT=DA
T+U: NEXT R: POKE DAT,MXPANT
 650 INPUT "NOMBRE DEL FICHERO:
"; LINE A$: IF A$="" THEN LET A$
="ED DATOS"
 660 CLS : PRINT AT 21,Z;"SALVAN
DO: ";A$
 670 SAVE A$CODE MAXBLO,FN D(DIS
P)-MAXBLO: GO TO 50
 680
 690 REM ** CARGAR DATOS **
 700
 710 INPUT "NOMBRE DEL FICHERO "
  LINE A$
 720 LOAD A$CODE : LET DAT=TABGR
F
 730 FOR R=MAXBLO+18 TO MAXBLO+3
4: POKE DAT,PEEK R: LET DAT=DAT+
U: NEXT R: LET MXPANT=PEEK R
 740 GO TO 50
 750
 760 REM ** GENERAR CODIGO **
 770
 780 CLS : PRINT AT 20,Z; INK 7;
 PAPER 2;" PULSA ENTER PARA GENE
RAR CODIGO    Y OTRA TECLA PARA
ABORTAR. "
 790 GO SUB KEY
 800 IF K$<>CHR$ 13 THEN GO TO 5
0
 840 CLEAR 32767
 850 LET I=USR 62615: LET J=PEEK
 23728+256*PEEK 23729
 852 PRINT "EL PROGRAMA OCUPARA
LA ZONA DE  MEMORIA: ";J;" A 655
35"
 854 IF I THEN PRINT "INCLUYENDO
: ";I;" HASTA ";J-1," (BUFFER DE
 PANTALLA)"
 860 INPUT "NOMBRE FICHERO: "; L
INE A$: IF A$="" THEN LET A$="ED
 COMPACT"
 870 SAVE A$CODE J,65536-J
 880 CLS : PRINT AT 19,0; INK 7;
 PAPER 2; FLASH 1;"    CARGA EL
CODIGO MAQUINA DE  NUEVO PARA VO
LVER A UTILIZAR    EL EDITOR (RU
N)                 "
 890 STOP
 900
 910 REM ** INICIA PARAMETROS **
 920
 930 GO SUB 3030
 960 INPUT "QUIERES CARGAR DATOS
 DE LA CINTAO ALMACENAR LOS GRAF
ICOS DE UNA PANTALLA (C/A) ? ";
LINE A$: IF A$<>"C" AND A$<>"A"
THEN GO TO 960
 970 IF A$="C" THEN GO TO 710
 971
 972 REM * ALMACENAR GRAFICOS *
 973
 999 SAVE "EP BASIC" LINE 0: SAV
E "EP C/M"CODE 62600,2936
1000 INPUT "CUANTOS GRAFICOS QUI
ERES           ALMACENAR (MAX. 255)
: ";MAXIMO: IF MAXIMO<U OR MAXIM
O>255 THEN GO TO 1000
1010 LET I=MAXIMO*4+FN D(TABGRF)
: RANDOMIZE I: POKE DISP,PEEK 23
670: POKE DISP+U,PEEK 23671
1020 POKE MAXBLO,Z
1025 INPUT "NOMBRE DEL FICHERO
           DE PANTALLA: "; LINE
 A$: LOAD A$CODE FN D(SCR),6912
1030 GO SUB 1870
1040 GO SUB KEY
1050 IF K$="C" THEN GO TO 1025
1060 IF K$="E" THEN GO TO 1130
1070 IF K$<>"I" THEN GO TO 1040
1080 IF PEEK MAXBLO=0 THEN GO SU
B 1970: GO TO 960
1090 POKE LASTB,U: POKE SCRTAB,P
EEK DISP: POKE SCRTAB+U,PEEK (DI
SP+U)
1092 PRINT #U;AT U,Z;"INTERFACE
KEMPSTON (S/N)            ": GO SUB K
```

También se muestra el número de pantallas creadas, la seleccionada en ese momento y la memoria disponible.

EDITANDO UNA PANTALLA

Desde el momento que pulsas "E" o "A", dejas el basic para entrar en el código máquina. La pantalla se limpia, los atributos pasan a ser papel azul, tinta amarilla y borde rojo (éstas son las condiciones por defecto, más adelante descubrirás cómo cambiarlas) y aparece una flecha, el cursor, en mitad de la pantalla. Siempre que la flecha esté presente en pantalla te encuentras en MODO CURSOR.

MODO CURSOR

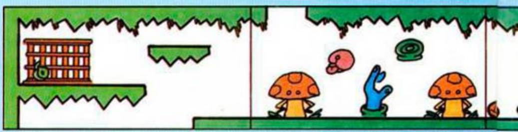
Dispones de las siguientes opciones:

"A" Cambiar atributos: un cuadrado parpadeante del tamaño de un carácter aparecerá en pantalla. Puedes moverlo utilizando los cursores (siempre que haya que mover algo deberás utilizar las teclas 5, 6, 7 y 8 y/o el joystick). En principio el atributo con que aparece es el definido como general, pero puedes cambiarlo pulsando dos teclas simultáneamente: una de ellas tienen que ser: "P"aper, "B"right o "F"lash y la otra una tecla numérica según el nuevo valor que quieras dar al papel, a la tinta, al brillo o al parpadeo. Con "C" copiarás el atributo así definido en la posición en que se encuentra el cursor, haciendo que este se mueva a la posición siguiente. Pulsa ENTER para salir aceptando los cambios hechos y BREAK para salir recuperados los atributos que había anteriormente. Este uso del

```
EY: IF K$<>"S" AND K$<>"N" THEN
GO TO 1092
1093 POKE KEMPS,A$="S"
1094 GO SUB 223: POKE SCRED,0: P
OKE NUMSCR,0
1098 INPUT "CUANTAS PANTALLAS VA
S A CREAR   (MAX. 255)? ";MXPANT
: IF MXPANT<U OR MXPANT>255 THEN
 GO TO 1092
1099 LET I=FN D(DISP)+MXPANT*2:
RANDOMIZE I: POKE DISP,PEEK 2367
0: POKE DISP+U,PEEK 23671: GO TO
 50
1100
1110 REM * ALMACENAR GRAFICOS *
1120
1130 LET POSY=Z: LET POSX=Z: OVE
R U: PAPER 8: BRIGHT 8: FLASH 8:
 INK 9
1140 LET DIMX=6: LET DIMY=24
1150 RANDOMIZE USR LDRSCR
1151 PRINT #1;AT 1,0;" Enter   Q
,A,P,O   5,6,7,8   F  "
1160 PLOT POSX,175-POSY: DRAW DI
MX*8-U,Z: DRAW Z,U-DIMY: DRAW U-
DIMX*8,Z: DRAW Z,DIMY-2
1170 GO SUB 1600
1180 IF K$="F" THEN GO TO 1030
1190 IF K$="O" THEN IF POSX>Z TH
EN LET POSX=POSX-8: GO TO 1150
1200 IF K$="P" THEN IF POSX+8*DI
MX<255 THEN LET POSX=POSX+8: GO
TO 1150
1210 IF K$="Q" THEN IF POSY THEN
 LET POSY=POSY-U: GO TO 1150
1220 IF K$="A" THEN IF POSY+DIMY
<176 THEN LET POSY=POSY+1: GO TO
 1150
1230 IF K$="5" THEN GO TO 1490
1240 IF K$="6" THEN GO TO 1390
1250 IF K$="7" THEN GO TO 1440
1260 IF K$="8" THEN GO TO 1340
1270 IF K$<>CHR$ 13 THEN GO TO 1
170
1280 IF PEEK MAXBLO=MAXIMO THEN
GO SUB 2030: GO TO 1090
1290 PLOT POSX,POSY: POKE 23728,
DIMX/2: POKE 23729,DIMY
1300 BEEP .2,20: RANDOMIZE USR S
TOGRF
1301 INPUT "": PRINT #1;AT U,Z;"
GRAFICOS ALMACENADOS: ";PEEK MAX
BLO
1305 IF INKEY$<>"" THEN GO TO 13
01
1307 GO TO 1170
1310
1320 REM ** EXPANDE DERECHA **
1330
1340 IF (POSX+DIMX*8>255) THEN G
O TO 1170
1350 LET DIMX=DIMX+2: GO TO 1150
1360
1370 REM ** EXPANDE ABAJO **
1380
1390 IF (POSY+DIMY=176) THEN GO
TO 1170
1400 LET DIMY=DIMY+U: GO TO 1150
1410
1420 REM ** COMPRIME ARRIBA **
1430
1440 IF DIMY=2 THEN GO TO 1170
1450 LET DIMY=DIMY-U: GO TO 1150
1460
1470 REM * COMPRIME IZQUIERDA *
1480
1490 IF DIMX=2 THEN GO TO 1170
1500 LET DIMX=DIMX-2: GO TO 1150
1510
1570 REM ** LEETECLA **
1580
1590 LET K$=INKEY$: IF K$<>"" TH
EN GO TO KEY
1600 LET K$=INKEY$: IF K$="" THE
N GO TO 1600
1610 RETURN
1620
1630 REM * PANTALLA PRINCIPAL *
1640
1650 BORDER Z: PAPER Z: INK 7: F
LASH Z: BRIGHT Z: OVER Z: CLS :
LET A$=" EDITOR DE PANTALLAS © J
.J.G.Q": GO SUB 1810
1660 PRINT ''': LET A$="DISENAR
PANTALLA": GO SUB LIN
1670 LET A$="SALVAR DATOS": GO S
UB LIN
1680 LET A$="CARGAR DATOS": GO S
UB LIN
1690 LET A$="VER PANTALLAS": GO
SUB LIN
1700 LET A$="GENERAR CODIGO": GO
 SUB LIN
1701 LET A$="FORMATO DE PANTALLA
": GO SUB LIN
1710 PRINT ''"PANTALLAS CREADAS:
 ";PEEK NUMSCR;"  "''"PANTALLA SE
LECCIONADA: ";PEEK SCRED;"  "''"M
EMORIA DISPONIBLE: ";FN D(MXDISP
)-FN D(DISP);"  "
1720 RETURN
1730
1740 REM * PONE LINEA DE MENU *
1750
1760 PRINT ;"  "; INVERSE U;A$(U
); INVERSE Z;"   ";A$''
1770 RETURN
1780
1790 REM ** CABECERA **
1800
1810 PRINT AT U,U; INK 7; PAPER
1;A$
1820 PLOT 4,171: DRAW 247,Z: DRA
W Z,-16: DRAW -247,Z: DRAW Z,16
1830 RETURN
1840
1850 REM ** SUBMENU ALMACENAR **
1860
1870 PAPER Z: INK 7: CLS : LET A
$="  ALMACENAMIENTO DE GRAFICOS
 ": GO SUB 1810
1880 PRINT ''': LET A$="CARGAR P
ANTALLA": GO SUB LIN
1890 LET A$="EDITAR GRAFICOS": G
O SUB LIN
1900 LET A$="IR AL MENU PRINCIPA
L": GO SUB LIN
1910 PRINT ''''"MEMORIA DISPONIBL
E: ";FN D(MXDISP)-FN D(DISP)
1920 PRINT "GRAFICOS ALMACENADOS
: ";PEEK MAXBLO
1930 RETURN
1940
1950 REM ** ERRORES **
1960
1970 LET A$="NO MAS ALMACENADO G
RAFICOS": GO TO ERR
1980 LET A$="NO EXISTE NINGUNA P
ANTALLA": GO TO ERR
1990 LET A$="NO PUEDES CREAR MAS
 PANTALLAS": GO TO ERR
2000 LET A$="EDICION ABORTADA":
GO SUB ERR: GO TO 50
2010 LET A$="NO HAY ESPACIO SUFI
CIENTE": GO SUB ERR: GO TO 50
2030 LET A$="NO PUEDES GUARDAR M
AS GRAFICOS": GO TO ERR
3000 BORDER Z: INPUT "": PRINT #
U; INK 7; PAPER 2; FLASH U;AT U,
Z;TAB (32-LEN A$)/2;A$,
3010 BEEP U,-10: GO SUB 1590: IN
PUT "": RETURN
3020
3030 REM ** INICIA VARIABLES **
3040
3041 POKE 23658,8: DEF FN D(X)=P
EEK X+256*PEEK (X+U)
3050 PAPER 0: INK 7: BORDER 0: C
LS : LOAD ""CODE 62600
3070 GO SUB 4000
3090 LET I=MKSCR-100: RANDOMIZE
I: POKE MXDISP,PEEK 23670: POKE
MXDISP+U,PEEK 23671
3100 RETURN
4000
4010 REM ** ETIQUETAS **
4020
4021 LET KEY=VAL "1590": LET ERR
=VAL "3000": LET U=SGN PI
4022 LET Z=NOT PI: LET LIN=VAL "
1760"
4050 LET R=VAL "65393"
4060 LET TABGRF=R
4061 LET SCRTAB=R+VAL "2"
4062 LET MODO=R+VAL "4"
4063 LET POSVEN=R+VAL "5"
4064 LET DIMVEN=R+VAL "7"
4065 LET SCR=R+VAL "9"
4066 LET ATTR=R+VAL "11"
4067 LET DISP=R+VAL "13"
4068 LET NUMSCR=R+VAL "15"
4069 LET SCRED=R+VAL "16"
4070
4071 LET R=VAL "39680"
4072 LET MAXBLO=R
4073 LET LASTB=R+SGN PI
4074 LET KEMPS=R+VAL "2"
4075 LET MXDISP=R+VAL "3"
4076
4077 LET R=VAL "62600"
4078 LET MKSCR=R
4079 LET ALTSCR=R+VAL "3"
4080 LET DELSCR=R+VAL "6"
4081 LET STOGRF=R+VAL "9"
4082 LET COPY=R+VAL "11"
4083 LET LDRSCR=R+VAL "13"
4085 LET PANTALLA=R+VAL "18"
4100 RETURN
9998
9999 SAVE "EP BASIC" LINE 0: SAV
E "EP C/M"CODE 62600,2936
```

LISTADO 2 EDITOR DE PANTALLAS

LINEA	DATOS	CONTROL
1	C318F5C333F5C364F818	1522
2	0A184218 75C32CFDC3A2	1090
3	FD3A009B3C32009BCD65	1037
4	FFED5BB05C73237223ED	1387
5	5B7EFF732372D5ED4B7D	1386
6	5CCD18FFD1ED4BB05CCB	1568
7	21C5E50600EDB0E1CD4B	1383
8	FEC110F3ED537EFFC93A	1666
9	81FFCD5DFFE53A80FF3C	1667
10	3280FF3281FFCD52FFED	1646
11	4B7EFF712370E1C5CD19	1368
12	FD280478C60347CD28F9	1183
13	D256F9E1EBED80ED537E	1864
14	FFC92A7AFF1100400100	957
15	1BEDB0C93E01320F9BAF	1099
16	32069B3E0E320E9BCD34	763
17	FE2A7EFF2323220A9B18	970
18	30AF320F9B3A81FFCD5D	1183
19	FF237E32069B2B7ECBBF	1190
20	320E9BCD19FDEBCD28F9	1431
21	D256F97ECB7FED5B7EFF	1710
22	EDB0ED530A9BCD34FE21	1442
23	8160220C9BF3AF3A0E9B	1071
24	1F1F1FE6073CCD9B223E	846
25	0832109B3E02CD0116CD	726
26	CCFBAFCD24FECD38F9CD	1840
27	18F7280E110808CD55FA	898
28	28F3ED430C9B18EACDB1	1394
29	FAD260F928E5CD9AFACD	1888
30	50F730DDFE07CA88F6FE	1698
31	422822FE58CA76F6FE4D	1379
32	CA40F6FE532866FE44CA	1515
33	50F6FE4120BDCD5CFECD	1622
34	8BF7CD24FE18A80100AC	1246
35	11090ECDC4FCCD62FE3A	1308
36	0E9B1814CD18F72825CD	971
37	9AFA573A099B5F7ACD23	1170
38	FC7BE67F32099B6F2600	1095
39	E53E03CD9CFBE1AFCDAD	1684
40	FB0640CDDAFBCDB1FAD2	1837
41	81F5FE0D20CCAF3A099B	1274
42	320E9BCD34FEC381F5CD	1504
43	14FADA8BF5DD6E02DD66	1528
44	03220C9BCD78F9C381F5	1347
45	CDC0FAD281F5AFCD24FE	1901
46	CD78F9C381F50100AC11	1333
47	180ECDC4FCCD62FE3E02	1312
48	CD9CFB2A039BED5B0A9B	1305
49	A7ED523E01CDADFBCDD7	1598
50	FBC381F5CD14FADA8BF5	1897
51	DD4E02DD4603CD64F7CD	1352
52	D7FBC381F53A069BA7CA	1623
53	5BF92A7EFF23773A0E9B	1144
54	2877CDC2F63A0F9BA720	1234
55	593A81FFCD52FF5E2356	1288
56	D5ED5B7EFF7228732A0A	1246
57	9B227EFFE1CD84F81853	1487
58	5F2A7CFF01000378BE20	865
59	07230B78B120F6C90100	830
60	03CD28F9D2F3F62A7EFF	1619
61	CBFEED5B0A9B2A7CFF01	1372
62	0003EDB0ED530A9BC906	1108
63	20CD9AFA10FBF1C381F5	1718
64	3A80FF3C3280FFCD52FF	1476
65	ED5B7EFF7323722A0A9B	1180
66	227EFFC35BF91E003EF7	1289



LINEA	DATOS	CONTROL
67	DBFECB672002CBDB3EEF	1536
68	DBFE0F0F2FE607B3C03A	1216
69	029BA7C8DBDFE60FC80F	1426
70	3002CBC30F3002CBDB0F	950
71	3002CBD30F3002CBCB7B	1058
72	A7C9FE31D8FE353FD8D6	1687
73	30473E00371710FD3210	594
74	9BC9C50100AC110E0ECD	976
75	C4FCCD62FE3E04CD9CFB	1683
76	C16948260044C53E00CD	940
77	ADFB3E2FD7E13E00C3AD	1403
78	FB2100583A0E9B32119B	821
79	7E32099B3A119BEE8077	1055
80	CD541FD226F83EBFDBFE	1542
81	1F3F387A3EFEDBFECB5F	1359
82	200A3A119B32099B3E01	549
83	180ACD4FF8282BCD18F7	1125
84	28D6012000E55FCB1BDC	1061
85	4CF8CB1BDC47F8CB1BDC	1543
86	3FF8CB1BDC37F8010040	1129
87	CDDAFBD13A099B1218A6	1313
88	CD2BF820CB15CD9AFA59	1450
89	1D01C70428181D01F801	576
90	28127AFE023095 1D01BF	854
91	072807017F081802CB22	453
92	10FC3A119BA1B232119B	1059
93	CD9AFAC398F73A099B77	1544
94	C9C511D2F80608CD54F8	1424
95	51C1C92B7CFE57C0265A	1303
96	C9097CFE5BC02658C9A7	1365
97	ED4218EC2318F111CAF8	1330
98	0604EB0E017EDBFE23A6	1060
99	2804230C10F5EBC93A81	975
100	FF4FCD52FF5E2356D554	1388
101	5D1B233A80FF3D3280FF	1090
102	3C918706004FC46FFEE1	1211
103	CD19FD2806040404 2424	613
104	24C5D5ED5B7EFFEBA7ED	1794
105	52444DE1EBE5C46FFEED	1714
106	537EFFD1C13A80FFDD2A	1570
107	73FFDD6E00DD6601A7ED	1429
108	52380919ED42DD7500DD	1034
109	7401DD23DD233D20E5C9	1152
110	DF01DF047F10FD08EF01	1095

LINEA	DATOS	CONTROL
111	F701F702F704F708F710	1266
112	EF10EF08FF1616011008	826
113	1108120813885052494F	520
114	5249444144BA4D454D4F	844
115	5249412044494 53504F4E	713
116	49424C45BA434F4C4F52	853
117	BA504F53582F59BA0001	839
118	7C0170017003480 14400	494
119	021000F8E5D52A0A9B09	924
120	ED5B039BA7ED52D1E1C9	1607
121	CD72FECD80FEED480C9B	1639
122	CD18FF3282FF545D2118	1153
123	F9010108CDB7FEC362FE	1448
124	21DA07180821320 01803	400
125	21D007FB22425CAF3244	984
126	5CED7B3D5CCDCCFBCD6B	1577
127	0DC3761B21CCFBE5DD7E	1417
128	01CD65FF5ECB23CB23CB	1335
129	232356ED53079BDDE5CD	1293
130	5CFEDDE1CD18F72815ED	1566
131	5B079BCD55FA28F2ED43	1379
132	0C9BDD7102DD700318DF	1086
133	CD81FAD27DFC28DECD9A	1840
134	FACD50F730D6FE0DC8FE	1765
135	43201C3A069B3C28C901	648
136	0400CD28F9D2F0F9CDA2	1564
137	FC110400DD19CDCCFB18	1203
138	ACFE502019CDE2FBAFCD	1625
139	24FE189F11E803210001	759
140	DDE5CDB503F3DDE11896	1702
141	FE582092ED4B0C9BCD64	1304
142	F7CDD7FBCDCCFBC38FF9	2165
143	DD2A0A9B3A069B3C4718	802
144	23DD7E01CD65FFED5B0C	1284
145	9B4ECB21CB21CB21DD7E	1288
146	02CD4CFA380A235A4EDD	1023
147	7E03CD4CFAD011FCFFA7	1559
148	DD1910D537C9BB28023F	1023
149	D8813DBBC9ED4B0C9B6F	1384
150	3A109B67CB1DDC91FACB	1382
151	1DDC8AFACB1DDC80FACB	1670
152	1DDC79FA2A0C9BA7ED42	1299
153	C97994D879944FC97884	1487
154	82FEC1D0788447C97894	1577

ENTER y de BREAK se puede extender, con estos mismos efectos, a casi todas las opciones de las que haya que salir en un determinado momento.

El programa está pensado para trabajar con pantallas monocromas; de esta forma no hay que almacenar los atributos de los gráficos o de la pantalla. Si tu cambias algún atributo, aún cuando sólo sea uno, los 768 bytes que componen los atributos se añadirán, al acabar la edición, a los datos de la pantalla. Es algo a considerar si tienes intención de crear bastantes pantallas (en 768 bytes se pueden almacenar los parámetros de 192 gráficos).

"B" Borra los atributos: realiza un CLS de los atributos con el color indicado por el valor númerico que aparece. Para cambiarlo se utilizan los cursores. El valor es el resultado de aplicar el habitual COLOR=BRILLO*64+PAPEL*8+TINTA. No se acepta un parpadeo general de la pantalla. ENTER o BREAK para salir.

"D" Memoria disponible: muestra la cantidad de memoria que queda.

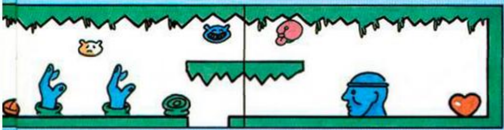
"M" Seleccionar un gráfico de memoria): en la esquina inferior izquierda aparecerá uno de los gráficos que existen en memoria. Usa los cursores para ver otros gráficos. ENTER sale con el gráfico seleccionado y entra en modo gráfico (ver modo gráfico). BREAK vuelve a modo cursor sin seleccionar ningún gráfico.

"S" Seleccionar un gráfico de los que existen en pantalla, se selecciona el gráfico que se encuentra bajo el cursor. Si existen varios gráficos siempre se toma el de mayor prioridad (el que se encuentre encima de todos) y se pasa a modo gráfico.

"X" Muestra las coordenadas X e Y del gráfico que hay bajo el cursor.

"5-6-7-8" (o joystick): para mover el cursor.

"1-2-3-4" Cambian la velocidad del cursor (en realidad es el número de pi-

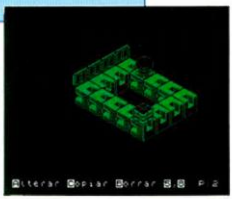


LINEA	DATOS	CONTROL
155	D8789447C9793D8483D8	1417
156	79844FC9F5E5D5C5DDE5	1867
157	1164002164000CDB503F3	882
158	DDE1C1D1E1F1C9CD541F	1835
159	D0AF32085CFF3A085CA7	1113
160	37C93A019BDD2A71FF3D	1162
161	26006F2929EBDD19CD45	986
162	FBCD18F728250F38030F	893
163	30143A009B4F3A019B3C	634
164	B92804380023E0132019B	556
165	18CE3A019B3D20F53A00	840
166	9818F0CD81FAD2CCF828	1756
167	CECD9AFAFE0D20C7DD46	1604
168	01DD2A0A9B3A019B0E00	657
169	DD3600FFDD7701DD7102	1207
170	3EC09047DD7703DD7104	1150
171	ED430C9BDDE5E1110400	1167
172	19220A9B3A069B3C3206	559
173	9B37C3CCFBDDE5CD72FE	1883
174	CD80FEAFCD24FEDDE1DD	1924
175	6E02DD6603DD5E003EB9	1000
176	DD960147DD560114140E	805
177	00C5CDC4FCC10404 79C6	1370
178	084FCD18FF3282FF3A01	1065
179	9BCDAAFEC362FE0100AC	1504
180	110E0ECDC4FCCD62FE3E	1317
181	01CD9CFBDD6E002600AF	1157
182	1811F53E013268 5CAFCD	978
183	A7F8F111E2F8C30A0CCD	1572
184	B6FB3E023268 5CC9E5D5	1389
185	1E20A7280C01F0D8CD2A	985
186	190118FCCD2A19C3361A	849
187	F5AFDBFEF6E03C20F8F1	1944
188	C90100000000008788120	542
189	F9C9DD7E0032099BCD72	1330
190	FEDDE5CD80FEDDE1CD83	2073
191	FBCD18F7280FCD9AFADD	1612
192	5E00CD23FC78CD40FC18	1254
193	DFCD81FA38063A099BC3	1334
194	40FC28DFCD9AFACD50F7	1720
195	30D7FE0D20D3C90F3004	1041
196	1CC01CC90F300678C60A	849
197	5F18F40F300778D60A5F	875
198	C01DC91D18FA57DD5E01	1128

LINEA	DATOS	CONTROL
199	DD4E02DD4603C5D5CD7D	1335
200	FCD1D50104000DD2A7EFF	1323
201	DD23DD23DD7E00A72809	1075
202	BA28023004DD0918F1CD	980
203	A2FCD1C1DD7200DD7301	1488
204	DD7102DD7003C9DDE5D1	1532
205	D5131313132A0A9BD5A7	876
206	ED5223444DE1D1EDB018	1373
207	ED530A9BAF123A069B3D	958
208	32069BC9DDE5D12A0A9B	1278
209	E5A7ED5223444DE1545D	1297
210	13131313ED530A9BEDB8	982
211	3A069B3C32069BC9D5C5	1101
212	D5CD18FFD1CD05FD0108	1378
213	10AFCD0AFD1520F9CD05	1171
214	FDC1CD3BFFC10C0C78C6	1500
215	020F0F0FE61F3C473A0E	511
216	9BC5E5545D77130D2804	953
217	0600EDB0E1C111200019	911
218	10EBC901F81F3EFFE5D5	1491
219	70237723 1D20FB71D1E1	1160
220	C34BFEE5E5236E260029	1206
221	29444D0303E109D11ACB	864
222	7FC9210058227CFF2100	895
223	40227AFF217FFDED5B7E	1342
224	FFA7ED52D57D2F4F7C2F	1376
225	47032A71FFEBA7ED52E5	1434
226	DD2A73FF3A80FFCDAEF8	1701
227	DD2A71FF3A009B08DD23	1108
228	DD233E01CDAEF8083D20	1047
229	F2DD2171FF3E02CDAEF8	1555
230	C103E1117FFDEDB813ED	1495
231	53B05C3A75FFA7C81E00	1178
232	7AD618474B3278FFE607	1174
233	3250FE78C618327DFFC9	1357
234	3A81FFCD5DFFE5DDE17E	1796
235	F5CD72FEDD7E01CD89FE	1762
236	F1CB7FDDE5E1E5CD34FE	1986
237	E13A75FFA72006CD62FE	1417
238	AF18572A76FF29292944	892
239	4DC5CD18FF7C92C64057	1377
240	5DED4B78FFCB20CB20CB	1453
241	20C5D5E50600EDB0E1CD	1520
242	4BFEEBE1CD45FEEBC110	1761

LINEA	DATOS	CONTROL
243	ECC1CD3BFFE57CE60367	1637
244	ED5B7CFF19D1ED4B78FF	1628
245	C5D5E50600EDB0012000	1091
246	E109EBE109EBC110EDC9	1585
247	ED5B7CFF21005801000 3	832
248	3801EBEDB0C9010003ED	1147
249	5B7CFF2005626B77130B	861
250	EDB0C9247CE607180624	1077
251	7CE607EE00C07DC6206F	1257
252	D87CD60867C9CD72FECD	1644
253	80FE2A7AFF7CFE40C811	1460
254	00400100018EDB0C92A7A	867
255	FF3600545D1301FF17ED	1021
256	B0C9DD2A7EFF3A069BA7	1407
257	C8DD23DD23F5CD9BFE11	1588
258	0400DD19F13D20F3C9DD	1249
259	4E02DD4603CD18FF3282	1038
260	FFDD7E01E5CD65FF4E23	1506
261	46237E23666FD1DDE579	1259
262	3C3C320CFF79873C3C3C	873
263	32F6FEC5D50600CDDAFE	1643
264	D1EBCD4BFEEBC110F0DD	1883
265	E1C9D5C5C5AF1183FFED	1848
266	B012133D1213C1EDB012	935
267	C1D1E53A82FFA7280F06	1302
268	092183FFA7CB1E2310FB	1130
269	3D20F20C41DD2183FF1A	1078
270	DDA605DDB60012DD2313	1088
271	10F3E1C978E60767781F	1296
272	1F1FE618B4677807 07E6	963
273	E059CB3BCB3BCB3BB36F	1389
274	ED5B7AFF1979E607C978	1409
275	E6C00707F65867780707	1007
276	E6E059CB3BCB3BCB3BB3	1508
277	6FC9ED5B73FF26006F2D	1204
278	2919C9CD52FF5E2356EB	1259
279	C9ED5B71FF26006F2D29	1132
280	2919C9249BCEAC000707	850
281	0C0C000800009836AE0404	540
282	0400000000000FFFFE7FFF	910
283	3FFFFFFF0042427E4242	1218
284	4200003E0808080 83E00	222
285	0002020242423C000044	266
286	4870484442000040 4040	582
287	40407E000042665A4242	644
288	420000426252 4A464200	522
289	003C4242424 23C00007C	508
290	42427C4040000003C4242	576
291	524A3C00007C42427C44	664
292	4200003C403C02423C00	378
293	00FE10101010100000 42	400
294	424242423C00F311FFFF	1094

DUMP: 40.000
N.° BYTES: 2.936



xels que se suman a la posición actual, en incrementos de 1, 2, 4 y 8). Es recomendable utilizar "4" para moverse por la pantalla y "1" para situar el gráfico con suficiente precisión.

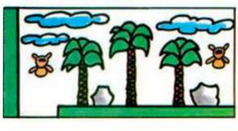
MODO GRÁFICO

Se accede a este modo seleccionando un gráfico de memoria o uno de los que existen en pantalla. Los opciones que hay son:

"C" Copiar el gráfico seleccionado en la posición actual.

"P" Cambiar la prioridad del gráfico. Utiliza el cursor para cambiarla. Si sales con ENTER se aceptará la nueva prioridad; si lo haces con BREAK continuará con la misma que tenía antes de entrar, aunque puede que no en las mismas condiciones. Para aclarar esto veamos cómo funciona la prioridad.

PRIORIDAD. Este valor, que en nuestro programa debe encontrarse entre 1 y 255, es algo parecido a una tercera coordenada espacial. La prioridad máxima es 255 y es asignada automáticamente a todo gráfico que se saca de memoria. Cada gráfico debe tener asociada una prioridad y, más aún, varios gráficos



LISTADO 3 CONVERSOR

```
10 REM ************************
       *   CONVERSOR DE       *
       *   GRAFICOS           *
       *   POR J.J.G.Q.       *
       ************************
20 PAPER 0: INK 7: BORDER 1: CLEAR : GO SUB 1000
30 INPUT "DIRECCION ALMACENAMIENTO ? ";DIRALM
40 INPUT "DIRECCION DEL GRAFICO: ";DIRGRF
50 INPUT "DIRECCION DE LA MASCARA: ";DIRMAS
60 INPUT "DIMENSIONES (X/Y) ";DIMX;" - ";DIMY: IF DIMX<1 OR DIMX>16 OR DIMY<1 OR DIMY>176 THEN GO TO 60
70 LET SCR=16384: CLS
80 FOR S=1 TO DIMY: LET PANT=SCR
90 FOR R=1 TO DIMX: POKE DIRALM,PEEK DIRGRF: POKE DIRALM+DIMX,PEEK DIRMAS: POKE PANT,PEEK DIRGRF: POKE PANT+DIMX,PEEK DIRMAS
100 LET PANT=PANT+1: LET DIRGRF=DIRGRF+1: LET DIRALM=DIRALM+1: LET DIRMAS=DIRMAS+1
110 NEXT R: LET DIRALM=DIRALM+DIMX: RANDOMIZE SCR: LET SCR=USR 23296: NEXT S
120 PRINT AT 20,0;"SIGUIENTE DIRECCION DISPONIBLE: ";DIRALM
130 GO TO 30
1000 RESTORE 1100: FOR R=23296 TO 23314: READ A: POKE R,A: NEXT R
1010 RETURN
1100 DATA 237,75,118,92,4,120,230,7,192,121,198,32,79,216,120,214,8,71,201

9999 SAVE "CONVERSOR" LINE 0
```

pueden tener la misma prioridad. En estos casos el orden de preferencia a la hora de ponerlos en pantalla es el de llegada: cuando se saca de memoria o se copia un gráfico con igual prioridad que otro ya existente; éste, el nuevo, se sitúa a continuación del último que tiene esa prioridad.

Si, por ejemplo, a un gráfico que se encuentra entre otros dos con igual prioridad, le cambias la prioridad y en el último momento te arrepientes (BREAK) el gráfico seguirá con la misma prioridad que tenía inicialmente, pero no en la misma posición dentro del grupo de gráficos con idéntica prioridad; ahora será el último.

GRÁFICO copiando el gráfico en la posición actual. Ten cuidado con esto porque es habitual copiar un gráfico en un punto y salir inmediatamente con ENTER, volviéndolo a copiar encima. BREAK abandona este modo borrando el gráfico seleccionado.

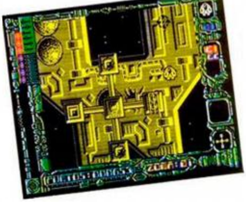
"1, 2, 3, 4" y **"5, 6, 7, 8"** Lo mismo que en el modo cursor.

Para volver el Basic puedes utilizar EDIT (la pantalla se almacena) o BREAK (la pantalla no se almacena). Si has alterado algún atributo y no dispones de suficiente memoria para almacenar todos los atributos, el programa generará un pitido y volverá al modo cursor para que hagas lo que creas más conveniente (sólo si sales con EDIT).

LISTADO 4 CONVERSOR

```
20 PAPER 0: INK 7: BORDER 1: CLEAR : GO SUB 1000: RESTORE 2000
40 READ DIRGRF
50 READ DIRMAS
60 READ DIMX,DIMY
65 READ POSY,POSX
67 PRINT AT POSY,POSX;
70 LET SCR=PEEK 23684+256*PEEK 23685
80 FOR S=1 TO DIMY: LET PANT=SCR
90 FOR R=1 TO DIMX: POKE PANT,PEEK DIRGRF: POKE PANT+DIMX,PEEK DIRMAS
100 LET PANT=PANT+1: LET DIRGRF=DIRGRF+1: LET DIRMAS=DIRMAS+1
110 NEXT R: RANDOMIZE SCR: LET SCR=USR 23296: NEXT S
130 GO TO 40
1000 RESTORE 1100: FOR R=23296 TO 23314: READ A: POKE R,A: NEXT R
1010 RETURN
1100 DATA 237,75,118,92,4,120,230,7,192,121,198,32,79,216,120,214,8,71,201
2000 REM ** AQUI LOS DATOS **
2010 REM ** EJEMPLO **
2020 DATA 40000,40120,4,10,0,0
2030 DATA 40240,40312,3,24,0,5
            |     |    |  |  | |
        DIRGRF    |    |  |  | |
            DIRMAS     |  |  | |
                    DIMX  |  | |
                       DIMY  | |
                          POSX |
                             POSY

9999 SAVE "CONVERSOR" LINE 0
```



Una buena costumbre para evitar este tipo de sorpresas es asignar prioridades diferentes a los gráficos que vas sacando de memoria. Puedes hacerlo incluso con incrementos de diez, para mayor seguridad.

"X" Útil para conocer las coordenadas actuales del gráfico que tienes seleccionado.

"ENTER" y **"BREAK"** ENTER abandona el modo

DETALLES TÉCNICOS

La estructura de datos utilizada es la habitual: existe una TABLA DE GRÁFICOS en la que cada gráfico tiene asociados cuatro bytes: dimensión horizontal (valor entre 1 y 32, baja resolución), dimensión vertical (entre 1 y 192, alta definición, incluidas las dos últimas líneas) y dirección de los datos que componen el gráfico y su máscara.

La otra tabla, la TABLA DE PANTALLAS (2 bytes por pantalla) contiene las direcciones donde comienzan los datos de cada pantalla.

Los datos que conforman una pantalla son: un byte destinado a indicar el atributo global, otro para especificar el número de gráficos que componen la pantalla, y los datos para cada uno de esos gráficos. Un gráfi-

co está representado por cuatro bytes: el primero indica la prioridad, el segundo el lugar que ocupa ese gráfico en la tabla de gráficos y los dos últimos las coordenadas, en alta resolución, donde se encuentra situado. Opcionalmente, si se alteró algún atributo, los atributos de toda la pantalla se almacenan a continuación de los datos del último gráfico.

El mapa de memoria es el siguiente:

• #8000 a #9AFF (32768 a 39679) se utiliza como buffer de pantalla.

• #9B00 a #9B23 (39689 a 39715) es una zona de variables.

• En #9B24 (39716) comienza la tabla de gráficos.

• A continuación de la tabla de gráficos y de sus datos comienza la tabla de pantallas.

• Al final de la tabla de pantallas se encuentra la zona de memoria disponible. Sus dimensiones variarán en función de la cantidad y volumen de gráficos que tengas almacenados. Antes de cargar ningún gráfico la memoria total operativa es de unos 22700 bytes.

• #F488 a #FF70 (62600 a 65392) el programa en Código Máquina.

• #FF71 a #FFC6 (65393 a 65478) Las siguientes variables:

— 65393/4 TABGRF: dirección de inicio de la tabla de gráficos.

— 65395/6 SCRTAB: dirección de inicio de la tabla de pantallas.

— 65497 MODO: indica el modo en que se presenta la pantalla; a 0 pantalla completa, a 1 con ventana.

— 65498/9 POSVEN: posición de la esquina supe-



Ejemplo de gráficos con sus respectivas máscaras en un mapeado de pantalla.

RESUMEN DE COMANDOS

MODO CURSOR

• A ALTERA LOS ATRIBUTOS.

5, 6, 7, 8: mueven el cursor.
C: copia el atributo seleccionado en la posición actual.
P+0—7: cambia el papel.
I+0—7: cambia la tinta.
B+0—1: cambia el brillo.
F+0—1: cambia el parpadeo.
ENTER: hace los cambios permanentes.
BREAK: se anulan todos los cambios hechos.

• B BORRA LOS ATRIBUTOS.

5, 6, 7, 8: cambian el valor del color.
ENTER: convierte el color seleccionado en global.
BREAK: no se realizan cambios.

• D MEMORIA DISPONIBLE.

• M SACA UN GRÁFICO DE LA MEMORIA.

5, 6, 7, 8: para ver los gráficos.
ENTER: selecciona gráfico y salta a modo GRÁFICO.
BREAK: vuelve a modo CURSOR.

• S SELECCIONA UN GRÁFICO de la pantalla y salta a modo GRÁFICO.

• X MUESTRA LAS COORDENADAS X/Y.

MODO GRÁFICO SELECCIONADO

• C COPIA EL GRÁFICO.

• P CAMBIA PRIORIDAD.

5, 6, 7, 8: para cambiar prioridad.
ENTER: acepta cambio.
BREAK: sin cambios.

• X IMPRIME LAS COORDENADAS X/Y.

• ENTER COPIA GRÁFICO Y VUELVE A MODO CURSOR.

• BREAK BORRA EL GRÁFICO Y VUELVE A MODO CURSOR.

PARA VOLVER AL BASIC

• EDIT ALMACENA LA PANTALLA CREADA.

• BREAK NO ALMACENA LA PANTALLA.

rior izquierda de la ventana.

— 65500/1 DIMVEN: dimensiones de la ventana.

— 65502/3 SCR: dirección del buffer de pantalla.

— 65504/5 ATTR: dirección de los atributos en el buffer de pantalla.

— 65506/7 DISP: primer byte disponible en la zona de memoria.

— 65508 NUMSCR: número de pantallas creadas hasta el momento.

— 65509 SCRED: pantalla seleccionada.

Cuando se utiliza la opción generar código el programa compacta los datos actualizando todas las direcciones y destruyendo la mayor parte del código del editor; dejando exclusivamente las rutinas implicadas en la impresión de la pantalla. Una vez compactados los datos no es posible realizar cambios en las pantallas. No olvides, por tanto, salvar los datos antes de utilizar esta opción.

Para imprimir una pantalla tienes que utilizar la rutina situada en 64930. Si al generar el código estabas en modo ventana se trabajará sobre un buffer de pantalla; si no, directamente sobre la pantalla. Para pasar de un modo a otro tienes que:

1) Alterar el contenido de MODO según el efecto que busques y 2) situar en las variables SCR y ATTR los valores adecuados: #4000 y #5800 para modo pantalla completa y las direcciones respectivas del buffer de pantalla si es modo ventana.

La pantalla que será impresa viene determinada por el valor que contiene la variable SCRED. Al no realizarse ningún tipo de comprobación sobre dicho valor, si éste no se encuentra en el rango adecuado (vale 0 o es mayor que el número de pantallas creadas) corres el riesgo de que el ordenador se bloquee.

Si en algún momento se interrumpe el programa no utilices RUN. Teclea únicamente GOTO 50 para volver al menú prinicipal. Si, a pesar de todo, lo haces, puedes reasignar las variables con GOSUB 4000 y dar a MXPANT el número máximo de pantallas que querías crear (si no lo recuerdas un valor menor que el original siempre será menos catastrófico).

Confiamos que esta utilidad satisfaga tus necesidades a la hora de planificar pantallas y, seguramente en menor medida (por la velocidad de impresión cuando la pantalla tiene muchos elementos), la de creación de juegos.

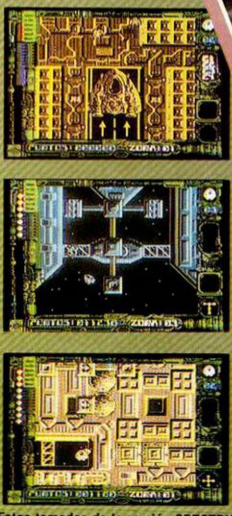
LISTADO EJEMPLO — No es necesario para que funcione el programa

LINEA	DATOS	CONTROL	LINEA	DATOS	CONTROL	LINEA	DATOS	CONTROL
1	14140124F4000E201CCD	600	60	00C2A30000FC00003FFF	927	119	4CFFF3220000000000FF	863
2	16AEA136050001002498	608	61	00328C0000FF0000FFFF	955	120	733FFCE2000000000FF	911
3	CEAC0007070C0C008000	544	62	000EF00000FFC003FFFF	1214	121	3CCFFB2200800000000FF	935
4	98E4AD030114053OA09B	945	63	0002C00000FFF00FFFFF	1214	122	0F33C8E200C0000000FF	939
5	052A809D0521249F0318	592	64	0000000000FFFC3FFFFF	1080	123	03CC8E2200F0000000FF	878
6	6EA00318FEA003188EA1	1041	65	0000000000FFFC3FFFFF	1080	124	0CB38FE200C0000000FF	1007
7	03181EA20318AEA20318	609	66	0003C00000FFF00FFFFF	1215	125	3FCC8FE2008000000FF	1019
8	3EA30318CEA303185EA4	906	67	000E300000FFE007FFFF	1058	126	7FB38FC20000000000FF	898
9	041CEEA4041CCEA5041C	869	68	000F900000FFE007FFFF	1155	127	7FCC8F020000000000FF	731
10	AEA6041C8EA7041C6EA8	991	69	001FC80000FFC003FFFF	1191	128	7FB38C020000000000FF	703
11	041C4EA9041C2EAA041C	559	70	001FE80000FFC003FFFF	1223	129	4FCC80020000000000FF	668
12	0EAB041CEEAB01010101	630	71	001FE80000FFC003FFFF	1223	130	73B380020000000000FF	679
13	01010101010101010101	10	72	001FE80000FFC003FFFF	1223	131	4CCC80020000000000FF	665
14	01010101010101010101	10	73	000FD00000FFE007FFFF	1219	132	3333800C008000001FF	626
15	01010101010101010101	10	74	000FB00000FFE007FFFF	1187	133	0CCC803000C000003FF	842
16	01010101010101010101	10	75	0013C80000FFC003FFFF	1179	134	033380C000F000000FFF	884
17	0000000000FFFFFFFFFF	1275	76	0018180000FFC003FFFF	1008	135	00CC830000FC00003FFF	905
18	0000000000FFFC3FFFFF	1080	77	002FF40000FF8001FFFF	1185	136	00338C0000FF0000FFFF	956
19	0003C00000FFF00FFFFF	1215	78	005BDA0000FF0000FFFF	1074	137	000CB00000FFC003FFFF	1148
20	000C300000FFC003FFFF	1020	79	00B7ED0000FE00007FFF	1056	138	0003C00000FFF00FFFFF	1215
21	00324C0000FF0000FFFF	891	80	0177EE8000F800001FFF	1020	139	0000000000FFFC3FFFFF	1080
22	00CEA30000FC00003FFF	939	81	06E7E76000F000000FFF	1074	140	000000FFFFDF000020FF	1020
23	033E54C000F000000FFF	851	82	0E6FF67000E0000007FF	969	141	FF0F0000F0FFFC070003	1027
24	0CFEAA3000C0000003FF	934	83	1797E9E800C0000003FF	1089	142	C8FFF003000F38FFC003	1219
25	33FE554C0080000001FF	850	84	10F66F0800C0000003FF	831	143	003CC8FF000300F328FC	1053
26	4C0000320000000000FF	381	85	1619986800C0000003FF	753	144	002303CC28F000A30F34	752
27	72AAFFC60000000000FF	992	86	1847E2D800C0000003FF	990	145	28E001A314C428C001A3	1040
28	3C54FF2A0000000000FF	696	87	18700ED800C0000003FF	819	146	1B1428D011A31A1428C8	761
29	6F2AFC6E0000000000FF	770	88	18BFFDD800C0000003FF	1137	147	D1A31A1428D0D1231A14	956
30	78C4F2AA0000000000FF	986	89	18BFFDD800C0000003FF	1137	148	C8C8D0C31A1530D0D107	1322
31	6EF2C6EE0000000000FF	1043	90	0DBFFDB000E0000007FF	1119	149	1A14C0C8D00F1A3700D0	950
32	68BC2AAA0000000000FF	762	91	0DDFFBB000E0000007FF	1149	150	303F1ACC00C8C0FF1A30	1062
33	6AEEEEEE0000000000FF	1075	92	0DDFFBB000E0000007FF	1149	151	00D003FF1AC000C80FFF	1154
34	6ABAAAAA0000000000FF	887	93	0EDFFB7000E0000007FF	1086	152	0B0000E03FFF000000F0	793
35	6AAEEEEE0000000000FF	1011	94	06EFF76000F000000FFF	1098	153	FFFF000000FFFFFF0000	1275
36	6AB6AAAA0000000000FF	883	95	076FF6E000F000000FFF	1098	154	00FFFFFF000000FFC3FF	1470
37	6AC6EEEE0000000000FF	1035	96	0B77EED000C0000003FF	1026	155	003C00FE3C7F01C380F9	1074
38	6806AAAA0000000000FF	708	97	3BB7EDDC0080000001FF	1083	156	C39F063C60F6006F0DC3	1081
39	6C06EEEE0000000000FF	845	98	79DBDB9E0000000000FF	972	157	B0EC00371A2458D8001B	860
40	7006AAAA0000000000FF	713	99	3CEDB73C0080000001FF	924	158	0C5A30EC18371624 68C6	825
41	6006EEEE0000000000FF	833	100	0E366C7000C0000003FF	738	159	006309C390E1C387063C	1068
42	3806AAAC0080000001FF	788	101	038661C000F000000FFF	936	160	60F03C0F09C390E80017	1014
43	0E06EEE000C0000003FF	932	102	00F00F0000FC00003FFF	825	161	143C28C400231900 98C1	721
44	0386AAC000C0000003FF	949	103	003FFC0000FF0000FFFF	1080	162	008336246C80240123C3	724
45	18E6EE1800C0000003FF	966	104	000FF00000FFC003FFFF	1215	163	C4800001 6B66D6080010	772
46	1E3EAC7800C0000003FF	834	105	0003C00000FFF00FFFFF	1215	164	36246C8000015EA57A40	772
47	118EE18800C0000003FF	970	106	0000000000FFFC3FFFFF	1080	165	81022766E4A0000519E7	921
48	1062C60800C0000003FF	770	107	0000000000FFFC3FFFFF	1080	166	98D8001B062460E60067	866
49	1418182800C0000003FF	558	108	0003C00000FFF00FFFFF	1215	167	01DB80F9DB9F003C00FE	1289
50	1506E0A800C0000003FF	869	109	000FF00000FFC003FFFF	1215	168	3C7F000000FFC3FF0000	892
51	15C283A800C0000003FF	964	110	003FFC0000FF0000FFFF	1080	169	00FFFFFF000000FF81FF	1404
52	15F28FA800C0000003FF	1024	111	00FFFF0000FC00003FFF	1080	170	007E00FC7E3F03FFC0F3	1260
53	15FABFA800C0000003FF	1080	112	03FFFFC000F000000FFF	1215	171	FFCF0FFFF0EFFFF71FFF	1999
54	15FABFA800C0000003FF	1080	113	0FFFFFF000C0000003FF	1215	172	F8DFFFFB3FFFF0BFFFFD	2246
55	15FABFA800C0000003FF	1080	114	3FFFDFFC0080000001FF	1177	173	3FFFFCBFFFFD5FFFFA5F	1964
56	167ABE6800C0000003FF	888	115	7FFF13FE0000000000FF	910	174	FFFA7FFFFE7FFFFE37FF	2087
57	119AB98800C0000003FF	942	116	7FFC1CFE0000000000FF	916	175	ECB7FFED7EFF7E7EFF7E	1925
58	0C62A63000E0000007FF	810	117	4FFF13320000000000FF	658	176	47DBE247DBE221FF84A1	1613
59	031A98C000F000000FFF	883	118	73FFDCE20000000000FF	1071	177	FF852CE734A0E705576A	1304

LINEA	DATOS	CONTROL
178	EA106A081785E8903409	1005
179	4B42D24842122A9954A8	954
180	8115153CA8D500A8067E	915
181	60E600670181800F9819F	1224
182	007E00FE7E7F000000FF	888
183	81FF000000FF81FF007E	1149
184	00FC7E3F03BDC0F3BDCF	1464
185	0F81F0EF81F7147E28D4	1397
186	0028387EDC88001D3800	720
187	DC88001D14002894 0029	682
188	2F81F48F81F1237EC483	1421
189	7EC12C7E34807E010F00	811
190	F0C000032F7EF4800001	981
191	337ECC800001387E1C80	848
192	00011800D8C000032BE7	713
193	D480000123E7C4800001	932
194	2CE73480000010F00F0C0	903
195	00030F7EF0E00007037E	744
196	C0F0000F007E00FC003F	888
197	000000FF81FF007E00FF	1020
198	81FF0381C0FC003F0C00	1035
199	30F0000F100008E00007	558
200	200004C0000320004C0	459
201	00034000028000015000	278
202	0A800001400002800001	334
203	4000028000012200 44C0	489
204	0003218184C00003107E	634
205	08E000070C0030F0000F	554
206	0381C0F8001F047E20E4	993
207	00271A0058D80001B3A00	454
208	5C88001D1D81B8DC003B	926
209	067E60E600670181 80F9	1068
210	819F007E00FE7E7F0018	945
211	00FF99FF000000FFE7FF	1404
212	000000FFFFFF003C00FF	1080
213	C3FF00C300FF00FF0100	1156
214	80FE007F0F00F0F0000F	1019
215	30DB0CC0180340240200	600
216	240040DB020018007100	458
217	8E0000003E007C800001	457
218	3D008C80000014D81B240	826
219	000271E78E70000E5CFF	961
220	3A5C003A373CECB700ED	979
221	4DC3B20DC3B0737ECE03	1284
222	7EC07CDB3E00DB003D3C	1063
223	BC803C0105C3A0C00003	932
224	01FF80F8001F00FF00FE	1172
225	007F001800FF00FF0000	661
226	00FFE7FF000000FFE7FF	1482
227	001800FF81FF006600FE	1019
228	007F018980F8081F0654	770
229	60E0540718AA98C0AA83	1250
230	35554C85554132AAAC82	1019
231	AAA139551C8155015E2A	852
232	7A402A026791E6601006	826
233	79E79E78001E1E7E789E	1094
234	00796799E66781E679E7	1421
235	9E79E79E1E7E789E7E79	1349
236	2799E48799E139E79C81	1506
237	E7811E7E78C07E030799	1117
238	E0E0180701E780F8001F	1118
239	007E00FE007F001800FF	786
240	81FF000000FFE7FF0000	1125
241	00FF81FF007E00FE007F	1146
242	01FF80F8001F07FFE0F0	1389
243	000F0FFFF0E000071DFF	1040
244	F8C0000031BFFF8C00003	1168
245	33FFFC80000137FFFC80	1377
246	000177FFFE0000007FFF	1011
247	FE00000077FFFE000000	882
248	7FFFFE0000007FFFFE00	1272
249	00007FFFFE0000003FFF	954
250	FC8000013FFFFC800001	1080
251	1FFFF8C000031FFFF8C0	1455
252	00030FFFF0E0000707FF	1006
253	E0F0000F01FF80F8001F	1142
254	007E00FE007F000000FF	762
255	81FF00000000FFFC3FFF	1209
256	0003C000FFF00FFF000E	974
257	7000FFC003FF00381C00	901
258	FF0000FF00E7C700FC00	1192
259	003F039FF780F000003F	903
260	0E7FFC00C000003F38FF	959
261	FB80800003BF38FFC100	1205
262	0000013F4E7FBB800000	584
263	3BBF739C11000000113F	618
264	4CFBBB800003BBBF5301	1107
265	10001001107F4CBBB800	623
266	0C3BBBFF329100000211	727
267	07FF0CBB8000C03BBFFF	1286
268	0390000 0F0107FFF00B8	969
269	0000FC3BFFFF00000000	821
270	FF07FFFF00000000FFFF	1282
271	FFFF00000000FFFFFFFF	1530
272	00000000FFFFFFFF0000	1020
273	0000FFFFFFFF00000000	1020
274	FFFFFFFF00000000FFFF	1530
275	FFFF00000000FFFFFFFF	1530
276	00000000FFFFFFFF0000	1020
277	0000FFFFFFFF00000000	1020
278	FFFE7FFF00018000FFF9	1524
279	BFFF0006C000FFE6DFFF	1607
280	0019E000FF99E3FF0067	1242
281	CC00FE67CDFF019F3600	1235
282	F99F36FF067CCF00E67C	1408
283	CF0F19F33E70D9F33E77	1305
284	3DCCF998BDCCF99B2633	1552
285	E67CA633E67D1ACF99FC	1564
286	C2CF99FD35EE67FC85EE	1824
287	67FD35319FFA85319FF8	1456
288	32D67FE280167FE035AF	1346
289	7F8A802F7F8031A9BE3A	1161
290	8029BE003596B8FA8000	1124
291	B80031ADA3FA8001A000	1108
292	358D4FFA800140003 1AC	937
293	BFF88000000 1198DBFF8	1173
294	C0000000301ACBFF0E400	1027
295	0007018DBFC0FC00000F	799
296	00CCBF00FE00003F000D	725
297	BC00FF2000FF000CB000	918
298	FFE003FF00064000FFF0	1302
299	0FFF00000000FFF9BFFF	1220
300	00000000FFFFFFFF0000	1020
301	0000FFFC3FFF0003C000	1020
302	FFE007FF001C3800FF80	1208
303	01FF00624600FE00407F	869
304	019E2980FC00283F037E	812
305	54C0F800541F06FC2A60	1035
306	F0002A0F0C018030E001	711
307	800715543FA8C1540003	751
308	26AA7F64A0AA00053354	905
309	7ECCB054000D5D8A79BA	1141
310	5C0A003A5E60467A5E00	636
311	007A6F9C39F66F8001F6	1178
312	6FE3C7F66FE007F677F8	1738
313	1FEE77F81FEE77FDBFEE	1706
314	77FC3FEE7801801E7800	1071
315	001E7AA9955E7AA8155E	969
316	3955AA9CB9542A9D0EA9	1119
317	9570CEA815730355AAC0	1221
318	F3542ACF00E99700FCE8	1444
319	173F0035AC00FF342CFF	917
320	000DB000FFCC33FF0003	957
321	C000FFF00FFF00000000	957
322	FFFC3FFF00000000FFFC	1332
323	3FFF0003C000FFE3C7FF	1449
324	001C3800FF9C39FF0061	904
325	C600FE60067F018AB980	1133
326	FD80018F032AAEC0F800	1235
327	00DF0655560F600006F	853
328	0B2AAED0EB0000D7128A	1041
329	B9D8D28001DB36E1476C	1417
330	B6E0076D2CDC3D74ACDC	1355
331	3D751D97E5B89D97E5B9	1493
332	5B76B6DA1B76B6D86764	1355
333	B3660764B36032ED988C	1245
334	82ED9B815CD9DD3A40D9	1520
335	DD02271B68E4A0186805	917
336	69E3C7966803C0165C3C	1154
337	3C325C0000325B47E2DA	858
338	5B4002DA336816CCB368	1039
339	16CD0E6D7670CE6D7673	1128
340	02CDBB40F2CDBB4F00D9	1388
341	BB00FCD9BB3F00339C00	1113
342	FF339CFF000ED000FFCE	1400
343	D3FF0002C000FFF2CFFF	1619
344	00000000FFFD3FFF0000	826
345	0000FFFC3FFF0003C000	1020
346	FFF3CFFF000E7000FFCE	1547
347	73FF00399C00FF381CFF	1177
348	00E66700FCE0073F0399	1035
349	99C0F38001CF0E67E670	1383
350	CE000073398C319CB800	907
351	001D3960069CB800001D	557
352	0E581A70CE0000733396	762
353	69CC838001C11CE5A738	1242
354	C0E0070323399CC48038	1054
355	1C0141CE7382000E7000	671
356	7A33CC5E0003C000341C	746
357	382C80000000147A3C5E2	886
358	40000002784182DE7800	726
359	001E347A5E2CB400002D	567
360	47B42DE2078001E03347	1004
361	E2CC834002C10C7BDE30	1225
362	C0781E0303342CC0F034	928
363	2C0F00C7E300FC07E03F	1031
364	0033CC00FF03C0FF000C	972
365	3000FFC003FF0003C000	948
366	FFF00FFF00000000FFFC	1272
367	3FFF00000000FFFC3FFF	1143
368	0003C000FFF3CFFF000C	1167
369	3000FFCC33FF0033CC00	1068
370	FF300CFF00CC3300FCC0	1269
371	033F0333CCC0F303C0CF	1161
372	0CCC3330CC0C30333333	732
373	CCCCB0300C0D3333CCCC	1167
374	B0300C0D1CCC3338DC0C	820
375	303B0733CCE0E703C0E7	1250
376	03CC3380FBC0039F03F3	1237
377	CE40F3F00E4F0BFC39B0	1342
378	CBFC39B337F7E7CCB7F7	1858
379	E7CD5FF0019A1FF01998	1374
380	67F0006607F7FE6059F8	1386
381	019A01FBF9805678066A	1102
382	0079E600559E19AA001E	819
383	18003567E6AC8007E001	942
384	0D599AB0C00180030356	845
385	6AC0F000000F00D5AB00	937
386	FC00003F0035AC00FF00	795
387	00FF000DB000FFC003FF	1149
388	0003C000FFF00FFF0000	960
389	0000FFFC3FFF00000000	825
390	FFFFFFFF00000000FFF7	1522
391	EFFF00081000FFEBD7FF	1478
392	00081000FF6BD6FF008C	995
393	3100FEADB57F009C3900	997
394	F69C396F08DDBB10EADC	1456
395	3B5709DE7B90E9DE7B97	1373
396	09DE7B90E9DE7B9715EE	1486
397	77A8D5EE77AB15AE75A8	1508
398	D5AE75AB0D4E72B0CD4E	1339
399	72B32E4812748E481271	890
400	6EC423760EC4237032E4	1094
401	274C82E427410CEC3730	928
402	C0EC3703132E74C8D02E	1121
403	740B1CCE73089C0E7039	871
404	5D324CBA1D0240B86DCC	997
405	33B60DC003B031D3CB8C	1220
406	81D00B810EDC3B70C0DC	1294
407	3B03031E78C0F01E780F	812
408	00EC3700FC0C303F0032	716
409	4C00FF0000FF000FF000	841
410	FFC003FF0003C000FFF0	1395
411	0FFF00000000FFFC3FFF	1095
412	00000000FFFC7FFF0003	892
413	8000FFF01FFF000FE000	1148
414	FFC007FF003C7800FF00	1144
415	01FF00F99E00FC01807F	1171
416	03FC6780F000601F0F1F	899
417	19E0E00018071E67C678	955
418	C06006033F19F1BC8018	966
419	01811FC67C7C80060001	742
420	67F19FFE6001800079FC	1355
421	6FCE780060007E7F1FB6	999
422	7E0000007F9FFF967F80	1072
423	00005FE7F3B65FE00000	1070
424	77F96D9677F800005DFE	1341
425	65965DFE0000577F6D86	1055
426	577F000065DF659A65DF	1117
427	0018397765B2B9770030	831
428	1E5D61CCDE5D00010797	898
429	66F0E797060301E56CC0	1263
430	F9E50C0F00797300FE79	1116
431	003F001EBC00FF9E80FF	1077
432	0007B000FFE783FF0001	1056
433	C000FFF9CFFF00000000	1158
434	FFFE3FFF00000000FFFD	1335
435	3FFF0002C000FFF00FFF	1277
436	000D3000FFC003FF0033	817
437	CC00FF0140FF00CFF300	1229
438	FC0AA03F033FFCC0F015	1256
439	540F0CFFFF30C0AAAA03	1204
440	33FFFFCC81555541 4FFF	1463
441	FFF20AAAAAA033FFFFCC	1772
442	81555416CFFFF3600AA	1206
443	AA0028 3FFCD480155401	974
444	484FF352400AA00272B3	1008
445	CFCE7001400E1CF42D38	977
446	9C000039672D74E20700	710
447	00E031CB539481C00381	1160
448	6A72CE5600700E004D9C	871
449	395A001C38005B47E6AA	793
450	0007E000336812CC8000	736
451	00010E6AB670C0000003	610
452	02DDB340F000000F00D5	934
453	9B00FC00003F0039CC00	731
454	FF0000FF000B7000FFC0	1080
455	03FF00034000FFF00FFF	1090
456	00000000FFFC3FFFAAAD	1168
457	58AD0AAD000000000000	444
462	000000000000000000E13	33
463	870A6F2E910A8136A50A	815
464	923FAF0AA348B90A9351	1052
465	C30A835ACD0A7363D70A	1080
466	646CE10A5575E10A5E36	1028
467	EB0A4C3FEB0A467EF50A	1080
468	3787F50A3A48FF0A2890	1024
469	FF0A2880FF0A2870FF0A	1115
470	2860FF0A29510414DC03	770
471	7837DC03883FDC039847	1043
472	DC03A84FDC03B857DC04	1188
473	7328DC088D36E603683F	978
474	E603A85FE6046330F003	1120
475	9867F0035847F0045338	1040
476	FF03484FFF035857FF03	1100
477	685FFF037867FF03886F	1185
478	FF04434 0FF098C66050E	915
479	C314A13ACD149042DC10	1105
480	7E4BE6106E53F0105D58	1080
481	FE0F1F4BFF0D3054FF0D	1043
482	405DFF0D5066FF0D5F6F	1081
483	FF0D6E78FF0E7F81FF11	1295
484	3C68FF123C5D00000000	593

DUMP: 39.680
N.º BYTES: 4.836





Fotos tomadas de la versión SPECTRUM

Ref. 10

SI NO LO ENCUENTRAS
EN TU TIENDA HABITUAL,
PIDELO AL CLUB ERBE,
NUÑEZ MORGADO, 11. 28036 MADRID.
TELEF. (91) 314 18 04.

SOFT

IEGOS
TURO,
Y





an sido necesarias para desarrollar
hasta ahora parecían imposibles
r, se han logrado estudiando,
icas de cada máquina, para
áximas prestaciones.

pixel, cuatro planos de fondo en
e profundidad como no has visto
randes diferencias que vas a
ADO" y "STARDUST"...

JEGOS DE NIÑOS!

DELEGACION BARCELONA. C/. VILADOMAT, 114 TELEF. (93) 253 55 60.

Pedro José Rodríguez Larrañaga

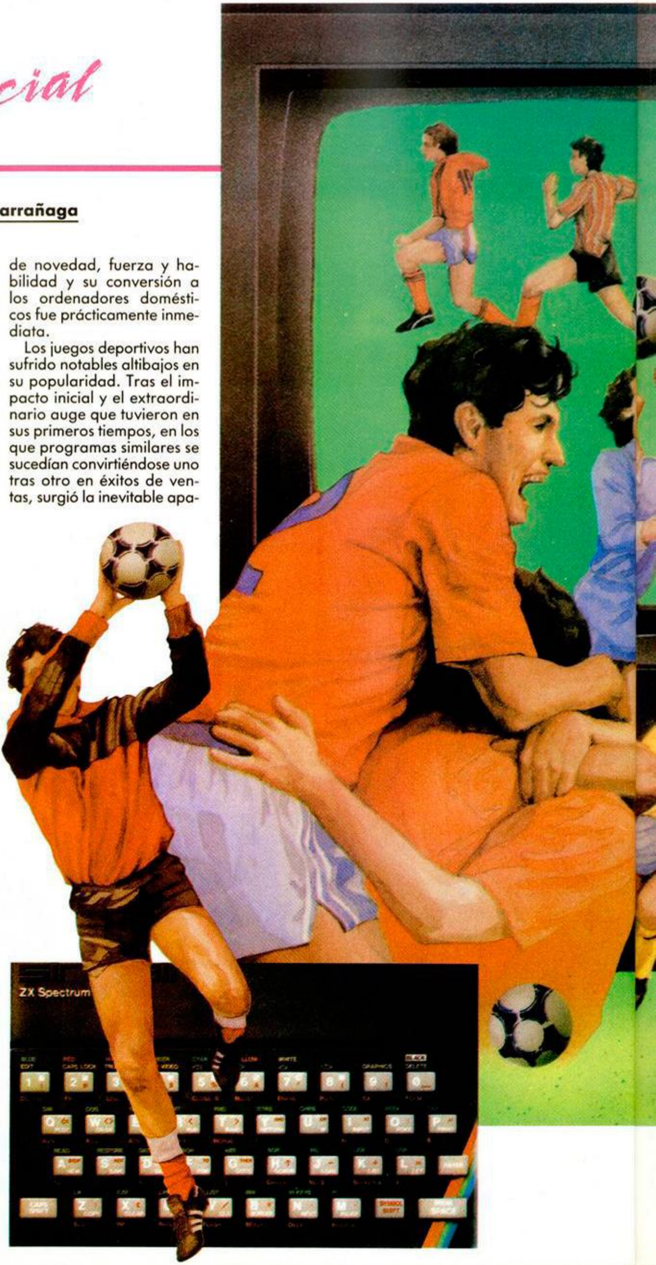
El deporte, hoy en día, ha ido mucho más lejos y se ha convertido en un fenómeno social que, sin duda, será estudiado en los libros de historia no dentro de mucho tiempo. Canalizador de frustraciones y deseos, el deporte se convierte en elemento de desafío personal para el que lo practica y en incentivo para salir de la rutina diaria para el aficionado. Hoy en día todos los deportes tienden, en mayor o menor medida, a convertirse en espectáculo de masas en torno a los cuales se mueven impresionantes cifras que dejan en ridículo los sueldos de otras profesiones.

Practicar un deporte, implica llevar una vida sana y nos ayuda a salir del stress diario de nuestra alocada vida moderna. Pero aún así hay quienes piensan que resulta más relajado simular una carrera con nuestro ordenador, cómodamente apoltronados en la butaca, que ponerse unas zapatillas y hacerlo de verdad.

Hablar de ordenadores es hablar del Spectrum, y el Spectrum debe gran parte de su popularidad a la avalancha de juegos deportivos que surgió no mucho después de su nacimiento. El fenómeno de los juegos deportivos arranca de la crisis que por esos momentos sufrían los arcades de acción, pues el adicto a las máquinas de los bares estaba bastante hastiado de la novedad que en su momento supusieron los «invaders» y demás sucedáneos.

Los primeros juegos deportivos, basados en pruebas de atletismo, tuvieron un éxito inusitado por la combinación que ofrecían de novedad, fuerza y habilidad y su conversión a los ordenadores domésticos fue prácticamente inmediata.

Los juegos deportivos han sufrido notables altibajos en su popularidad. Tras el impacto inicial y el extraordinario auge que tuvieron en sus primeros tiempos, en los que programas similares se sucedían convirtiéndose uno tras otro en éxitos de ventas, surgió la inevitable apa-





El deporte ha acompañado a la humanidad desde sus comienzos como imagen y reflejo de sus deseos y pasiones. Detrás del deporte se esconden constantes humanas tales como el afán de superación, la competitividad, la amistad y el sacrificio, la pasión y la rivalidad. Como en un sueño sin retorno el deportista sólo encuentra satisfacción en la automejora y la victoria. Pero, como siempre, esta victoria sólo puede corresponder a uno.

el Deporte en el Spectrum

tía y hubo un corto espacio de tiempo en el que se dejaron de lanzar nuevos productos. Actualmente los juegos de este tipo han vuelto con un empuje más moderado, de forma que de una manera más o menos continua aparecen títulos de este tipo en el mercado, programas que intentan en la medida de lo posible ofrecer nuevas posibilidades y explorar terrenos desconocidos en el campo del ordenador personal.

La relación que encontraréis a continuación pretende ser una guía más o menos sistemática de los programas deportivos que han pasado por el mercado del Spectrum desde sus inicios, hace ya casi cinco años. Nuestra intención es por un lado ofrecer una sana crítica de los programas que hemos considerado más representativos dentro del género y por otro lado elaborar una lista agrupada por especialidades en la que se informe detalladamente al comprador de juegos de los programas existentes en su especialidad favorita. Tened en cuenta que la mayoría de los programas comentados están ya descatalogados por su antigüedad y se han convertido en clásicos de la programacion.

En cualquier caso, como siempre, la elección está en vuestras manos.

TENIS Y DERIVADOS

Hablar de tenis es hablar del maravilloso Match Point de Psion, casa de Software que se convirtió en la pionera a la hora de elaborar programas comerciales para el Spectrum. Desde la inolvidable cinta de demostración «Horizontes» hasta las aventuras de Horacio, pasando por los Space invaders y las serias aplicaciones de la serie VU, Psion demostró un conocimiento del ordenador impensable por el escaso tiempo transcurrido desde su lanzamiento.

• **Match Point** es, pese a su antigüedad, el mejor programa de tenis para Spectrum. Tras un cuidadoso menú que ofrece programar nivel de juego, controles y duración del partido, el juego en sí es una sofisticada simulación en la que la sombra de la pelota, el público que mueve la cabeza, la excelente perspectiva y el perfecto control que dispo-

nemos sobre la trayectoria de la pelota le convierten en un juego terriblemente adictivo. El ordenador es prácticamente imposible de vencer en los niveles altos y el único defecto que se le puede achacar es la simplicidad de los gráficos de los jugadores.

• **Tennis,** de Imagine, es una versión bastante mediocre del programa original realizado por Konami para MSX. Tennis aporta como novedad jugar partidos de dobles y los gráficos son notablemente más detallados, pero el juego es demasiado lento, el control de la pelota prácticamente nulo y los colores empleados difíciles de distinguir.

• **Ping Pong,** de Imagine, es un programa excelente, cuidado al máximo. Es además la única simulación para ordenador de un partido de tenis de mesa. Un sonido espectacular acompaña a un movimiento preciso y rápido, a la vez que presenta un sistema de controles muy práctico aunque difícil de asimilar al principio.

Acabamos este capítulo con el **Squash** de New Generation, programa que pasó verdaderamente desapercibido en su momento pese a la novedad de ser el primer simulador de dicho deporte. Sorprende al jugador con una pantalla demasiado pequeña y unos gráficos esquematizados y simples, pero pasando por alto estos detalles visuales el programa puede ofrecer unas moderadas dosis de diversión por lo rápido y frenético de su desarrollo.



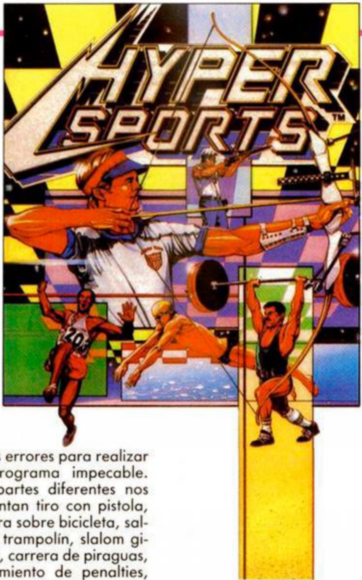
ATLETISMO

Los juegos sobre pruebas atléticas, también conocidos como juegos de olimpiadas, son sin duda el estereotipo del género que los engloba y los que más diversión y buenos recuerdos ofrecen al aficionado. Los juegos de este tipo han sido causantes de muchos teclados machacados y dedos entumecidos en el noble arte de golpear sin descanso las teclas de nuestro querido Spectrum (las tan famosas teclas de goma tenían que tener alguna ventaja), y han sido muchas las tardes transcurridas en largas series de competiciones buscando mejorar la marca personal o aguantar un mayor número de fases.

• **Decathlon** de Ocean fue el detonante de lo que sería una larga serie de programas basados en diferentes pruebas atléticas. Las novedades que introdujo fueron muchas, desde el entonces revolucionario sistema de carga turbo hasta presentar las diez pruebas del decathlon divididas en dos partes que habían de ser cargadas individualmente. Con la colaboración del plusmarquista mundial de decathlon, el inglés Daley Thompson, el programa ofrece las pruebas que conforman dicha especialidad: 100 metros lisos, salto de longitud, lanzamiento de peso, salto de altura, 400 metros, 110 metros vallas, pértiga, jabalina, martillo y 1.500 metros. Los primeros ciclos son verdaderamente fáciles, pero las marcas exigidas aumentan cada fase hasta exigir registros imposibles. El gráfico del jugador deja algo que desear y el perfecto movimiento se contrarresta con una excesiva lentitud.

Pero el mejor juego de olimpiadas es sin duda **Hypersports.** Versión, al igual que el Decathlon, de una máquina recreativa diseñada previamente por Konami. Se dice que Hypersports constituyó en su día un récord de originales vendidos en nuestro país, un país en el que las ventas de originales suponían un porcentaje mínimo sobre el total. El programa abarca en un solo bloque seis pruebas: natación, tiro al plato, tiro al arco, salto de potro, triple salto y halterofilia. Una sola vida y marcas altas desde el primer momento, excelentes gráficos y colorido y un movimiento preciso y rápido le convirtieron en un merecido éxito que indirectamente, hizo muy difícil mejorar la calidad exhibida lo que condujo en cierta manera a la decadencia de este tipo de juegos.

Algo posterior al Hypersports, de Imagine, y continuación directa de la línea marcada por Decathlon, el **Supertest**, de Ocean, vuelve a utilizar la imagen de Daley Thompson para realizar un programa notablemente más ambicioso, en el que se corrigen algunos an-



tiguos errores para realizar un programa impecable. Dos partes diferentes nos presentan tiro con pistola, carrera sobre bicicleta, salto de trampolín, slalom gigante, carrera de piraguas, lanzamiento de penalties, salto de esquí y prueba de cuerda. Como podéis ver pruebas originales y divertidas, pero con una diferencia demasiado acusada de dificultades lo que hace que la adicción fluctúe a lo largo de la partida. Detalles graciosos y una estupenda puesta en escena le convierten en un excelente programa.

No podemos dejar de mencionar al **Videolimpic**, de Dinamic, aunque sólo sea como reconocimiento al primer programa deportivo que se hizo en nuestro país. Notablemente inferior a los juegos ya comentados, incluye los 100 metros lisos, longitud, jabalina, martillo, 110 vallas y natación. A destacar el rápido movimiento conseguido mediante el scroll por caracteres en vez de por pixels y los simpáticos rótulos publicitarios, en los que MICROHOBBY tuvo su hueco, pero la parte intrínseca del juego referida a regular velocidades o ángulos de tiro ofrece escasas posibilidades. Un detalle más en un programa cargado de ellos: con el título de Videolimpics y editado por Mastertronic, dentro de su línea de dos libras, este juego ha conseguido ser muy recientemente número uno en Inglaterra, lo cual no deja de ser anecdótico teniendo en cuenta que se trata de uno de los peores juegos de la compañía española.

• **Sports hero**, de Melbourne house, salió al mercado en plena fiebre de juegos de olimpiadas y no tuvo una acogida demasiado favorable en nuestro país, al resultar bastante inferior a programas similares introduciendo mínimas diferencias como ambientar cada fase en escenarios diferentes (la calle, la universidad y el estadio). Parece confirmarse la aparición de la versión para Spectrum del Summer Games (juegos de verano) de la casa Epyx, avalado por su extraordinario éxito en la ya lejana versión para Commodore.

JUEGOS DE LUCHA



No es preciso romperse la cabeza para explicar la razón del tremendo éxito de los juegos de lucha, los programas deportivos que se han venido produciendo ininterrumpidamente hasta nuestros días. Reconocer el componente de violencia que contiene nuestra cultura es un hecho al que no nos podemos mantener ajenos.

Los primeros pinitos en los juegos de lucha vienen de la mano de las artes marciales. Uno de los primeros juegos que me viene a la mente es el **Kung-fu**, de Bugbyte, ambientado en un gimnasio oriental en el que orondos luchadores combatían cuerpo a cuerpo contando con un aceptable número de movimientos posibles, si bien todo el conjunto del programa quedaba ensombrecido ante la lentitud de los movimientos.

Dinamic demuestra estar «al loro» en la recién estrenada disciplina y crea lo que sería el primer simulador de boxeo al que denominaría, por supuesto, **Rocky**. El programa tuvo cierto éxito en su momento y posee unos gráficos asombrosos, de gran tamaño y detalle, pero para el usuario actual pierde validez pues los luchadores se mantienen estáticos, limitándose a golpear o defenderse con uno o ambos puños. Además, el juego resultaba demasiado fá-

cil, era suficiente un par de horas de práctica para alzarse con el título de campeón mundial.

Sin embargo, el mercado del software se ve completamente revolucionado con lo que acabaría convirtiéndose en el mejor simulador de lucha de su época, y para algunos el mejor programa de sus características de todos los tiempos. Nos referimos, por supuesto, al extraordinario **The way of the exploding fist**, la obra maestra de Melbourne House. Los adjetivos son pocos para hacer honor a las virtudes de un programa en el que los gráficos, sonido, rapidez de movimientos, instantánea respuesta del teclado, dificultad media y variedad de golpes le convierten en un juego que, como el tiempo nos ha demostrado, ha sido muy difícil de mejorar. Tal vez el único juego que ha alcanzado su categoría haya sido **The way of the tiger**, de Gremlin graphics, nacido bajo la estela del anterior pero incorporando gran cantidad de detalles en el decorado, variedad de enemigos, tres diferentes combates a cargar de la cinta y un revolucionario sistema de triple scroll, en el que la pantalla se divide en bandas laterales que se desplazan a diferentes velocidades, consiguiéndose una gran sensación de realismo.

Encontramos a continuación programas que no aportan demasiado, frente a los títulos ya citados. **International karate**, de System 3, es un nuevo simulador de lucha muy en la línea del Exploding fist que introduce como novedades voz digitalizada y varios escenarios correspondientes a diferentes partes del mundo, que el jugador recorre a medida que vence adversarios, apareciendo por vez primera un esbozo de argumento frente a la mera mecánica de los combates. **Frank Bruno's boxing**, simulador de boxeo, se muestra muy inferior al Rocky en cuanto a gráficos, pero en él, el boxeador se mueve con cierta libertad por el ring y se permite una mayor variedad de golpes.

Por desgracia, el mundo de los simuladores de lucha se encuentra mucho más repleto de medianías que otros campos del software deportivo. Bajo nuestros atónitos ojos hemos contemplado engendros como el Kun-fu master, el Rock'n wrestle o el Ninja master. El primero, el mayor patinazo jamás realizado por Melbourne House, presenta gráficos a base de pixels dobles y unos movimientos que dan pena de los malos que son, mientras que el jugador lucha en vano por conseguir que su luchador responda a sus movimientos. El segundo, una de las acostumbradas castañas con las que nos suele obsequiar US Gold, asombra al personal por un machaque de atributos y unos gráficos malos con avaricia y se entronca en los incontables juegos «chico valiente rescata a chica raptada». El tercero, bajo el sello de la serie plata de Firebird, se encuentra en la línea de los más clásicos juegos de olimpiadas, sustituyendo los eventos olímpicos por pruebas más orientales y exóticas, como romper bloques a base de golpes de karate o interceptar shurikens. El juego crea cierta adicción pero los gráficos del personaje y decorados lo echan todo a perder (curiosidad: se cuelga si está conectado el Interface I). Es una lástima que los dos primeros hayan salido tan mal parados frente a las excelentes versiones para otros ordenadores.



Hay otros muchos títulos de calidad con grandes similitudes con los ya comentados, como pueden ser Sai combat (Mirrorsoft), Yie ar kung-fu (Imagine) o Shaolin's road (The edge).

Hemos excluido de nuestro estudio aquellos programas que, teniendo como elemento primordial la lucha, añaden a la acción objetivos a conseguir o elementos de videoaventura, aunque la mayoría de las veces nos limitemos a rescatar a la princesa de turno. En este grupo la lista sería interminable; por citar a algunos nombremos Fighting warrior y Fist 2 (Melbourne house), Yie ar kung-fu 2 (Imagine), Ninja (Mastertronic) y el mismísimo Barbarian (Palace), uno de los mayores éxitos de la actualidad.

FÚTBOL



Dicen que el fútbol es el deporte rey, lo que no estoy dispuesto a admitir es que digan que es el rey de los deportes. Fenómeno social que casi nadie se ha molestado en estudiar con la atención necesaria, se convierte en eficaz medio de apaciguar a las masas y evitar que recuerden sus propios y más importantes problemas.

El primer intento serio de hacer un programa de fútbol para Spectrum viene de la mano del **World cup**, de Artic. Como en todos los programas que irán surgiendo tras él, el campo no cabe en la pantalla, por lo que ésta va scrollando lateralmente. Esta versión no dispone de la posibilidad de mover a los porteros, resulta bastante lenta y carece del realismo necesario, pero hay que reconocer su validez para la época en que surgió y detalles interesantes como que el juego simula un torneo de copa del mundo (de ahí su nombre), en el que pueden jugar varios jugadores escogiendo cada uno un país o un sólo jugador permitiendo que el ordenador elija aleatoriamente los países hasta llegar a la final.

El primer programa de fútbol de indiscutible calidad es el ya mítico **Match Day**, de Ocean, que supuso en su momento un verdadero exitazo por sus excelentes características. Un exhaustivo menú ofrece la posibilidad de cambiar todo lo cambiable, desde el color del campo o jugadores hasta los nombres de los equipos, la duración de los partidos, los controles a utilizar y el sistema de juego, que permite la elaboración de una liguilla en la que pueden competir hasta ocho jugadores (imaginad una tarde entera montando una liga cómodamente en casa con tus amigos). En el plano meramente técnico, el programa se revela un pelín lento, pero es inevitable dado el gran número de sprites en pantalla. Movimientos muy racionales, excelente sensación de tridimensionalidad, el bote y la sombra del balón y la posibilidad de mover el portero le convierten en un programa completo y fácil de manejar a pesar de las innumerables opciones que presenta.

De los mismos autores y para demostrar que todo es mejorable ha aparecido, hace relativamente poco tiempo, el **Super soccer** de la mano de la misma casa de software. Las innovaciones afectan al reconocimiento del jugador que está siendo controlado, el cual lleva una visible coronilla en vez de distinto color como en el programa anterior, y a la inclusión de marcadores que indican velocidad, energía y fuerza de los jugadores, a la vez que incluye la posibilidad de lanzamiento de faltas, corners y penalties. Los gráficos son similares y se respetan y mantienen todas las posibilidades anteriores, por lo que el conjunto queda sensiblemente mejorado a la vez que se garantiza la fácil adaptación de los adictos al Match Day.

• **Peter Shilton's handball Maradona**, de Gremlin, es un curioso «simulador de portero» con el que puedes intentar dar la vuelta a la histórica final del Campeonato del mundo Argentina-Inglaterra. Permite escoger entre tres niveles (práctica, competición y mejora) y la posibilidad de modificar la habilidad de tus oponentes. Si escoges la opción de dos jugadores podrás optar entre actuar de portero o de atacante.

Hemos dejado para el final lo que es, sin duda, el mayor timo jamás visto en la historia del software. La criatura, lo habréis imaginado, se llama **World cup carnival**, y el padre de la criatura, claro está, no puede ser otro sino U.S. Gold. Lo irritante del caso no es que el programa fuera malo y caro para los tiempos que corrían sino que se engañó vilmente a los pardillos que lo compraron, cogiendo el ya comentado World cup, cambiando los mensajes de pantalla, incluyendo tres pruebas sosas y aburridas, colocando cuatro pegatinas y un póster, y vendiéndolo como juego oficial del campeonato del mundo. El programa se vendió mucho, lo que demuestra que es bastante fácil engañar a la gente, sobre todo, cuando está desinformada. Un escándalo que esperemos que no se repita por bien de la credibilidad del software.

BEISBOL

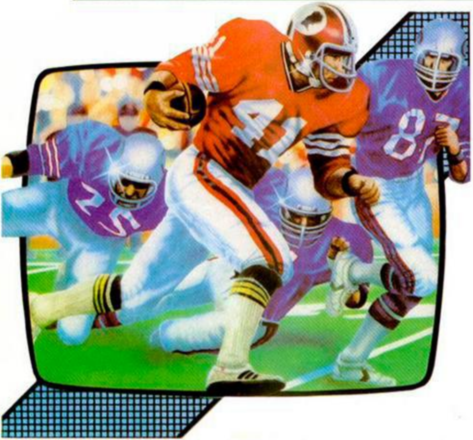
El baseball, deporte que en España cuenta con una difusión prácticamente nula, es, sin embargo, uno de los grandes espectáculos deportivos de los Estados Unidos, país en el que se le puede considerar casi el deporte nacional. Juego de equipo en el que las individualidades tienen mucha mayor importancia que en cualquier otra disciplina, el baseball combina la fuerza, la habilidad y la rapidez en la carrera todo ello aderezado con una importante dosis de violencia. Los juegos de baseball para Spectrum han contribuido, por lo menos, a conseguir que los aficionados españoles tomaran contacto con él y conocieran sus técnicas y reglas.

El programa más conocido de esta especialidad es, sin duda, el ya antiguo **World series baseball**, uno de los primeros lanzamientos de Imagine tras su integración en Ocean. Aparecido en plena fiebre de los juegos deportivos el programa se convirtió en una gran éxito de ventas. La pantalla muestra el campo desde una perspectiva frontal. Una gran pantalla amplía la figura del bateador y muestra diversos mensajes referidos al transcurso técnico del partido y hasta mensajes publicitarios. El jugador puede controlar la fuerza e inclinación de la pelota, así como al corredor en determinadas circunstancias. Programa adictivo y simpático, de fácil comprensión y control, tiene su único fallo en unos gráficos de jugadores y majorettes pequeños y simples, pero que se suplen con la gran diversión que aporta. Por lo demás, el juego es totalmente real y se ajusta perfectamente a las reglas del deporte.

Bastante más reciente es el **Hardball**, de Accolade, versión para Spectrum de uno de los mejores programas jamás lanzados para Commodore. Las novedades aparecidas son muchas pues la filosofía del programa se aparta bastante del World series. Mientras que en éste teníamos siempre una visión total y panorámica del estadio, en esta ocasión la pantalla recoge con mayor detalle la escena en la que transcurre la acción, ya sea el bateador o los corredores. El resultado es una excelente representación gráfica gracias al exquisito uso del color y el gran tamaño de los sprites, pero tales derroches artísticos provocan una excesiva lentitud de movimientos que, en un programa de estas características, impide al jugador meterse de lleno en la tensión del partido. Un programa completamente diferente al anterior que seguro que agradará a muchos.



RUGBY

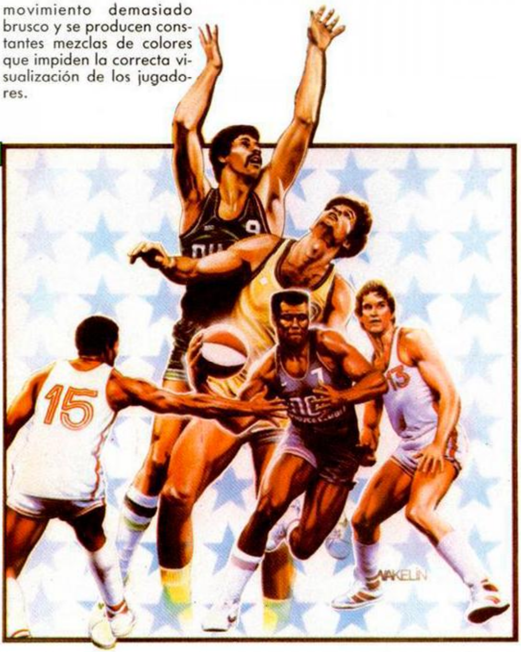


Otro deporte con escasa popularidad en nuestro país y sin duda uno de los más violentos y duros. Mantiene la curiosa mezcla de estrategia y rudeza física, siendo el deporte grupal con mayor índice de accidentes y daños físicos entre sus practicantes. Se juega con un curioso balón ovalado en un campo perfectamente señalizado y tiene la también curiosa particularidad de ser el deporte en el que más tiempo se mantiene el balón parado en posesión del árbitro a raíz de las diversas infracciones que cometan los equipos.

La representación de este deporte para Spectrum es casi nula y de escasa calidad, teniendo su primer exponente en el **American football**, de Mind games, un curioso programa de gráficos terriblemente esquematizados representados desde una vista aérea vertical. El campo aparece señalizado como en la realidad y los jugadores son puntitos alocados que corretean por un césped rojo (curioso, ¿verdad?), de modo que el interés del programa se centra en su componente estratégico. Otro programa que pasó sin pena ni gloria fue el **International rugby**, de Artic, programado y diseñado de una manera prácticamente idéntica

al World cup, lo que, evidentemente, no dice mucho en su favor. El juego demuestra haber sido realizado con buena intención y puede llegar a ser divertido, pero los jugadores tienen un movimiento demasiado brusco y se producen constantes mezclas de colores que impiden la correcta visualización de los jugadores.

BALONCESTO



El baloncesto ha protagonizado una de las más vertiginosas escaladas de popularidad de los últimos años. Si bien el prestigio de dicho deporte en España arranca desde muy atrás, las últimas temporadas y los cambios introducidos en el sistema de juego han hecho del deporte de la canasta el segundo en aceptación en nuestro país, detrás del inevitable fútbol. Debo confesar mi particular preferencia por un deporte que, personalmente, considero uno de los más completos y apasionantes que existen, un juego en el que el tiempo se apura hasta el último segundo, en el que las acciones de ataque y defensa se suceden rápidas y trepidantes, en el que a veces no se sabe el ganador del partido hasta sus últimos instantes. El baloncesto es movilidad y espectáculo, carrera contra reloj en el que la cohesión del equipo se muestra imprescindible como en ningún otro, donde cualquier brote de violencia se ataja desde el primer momento.

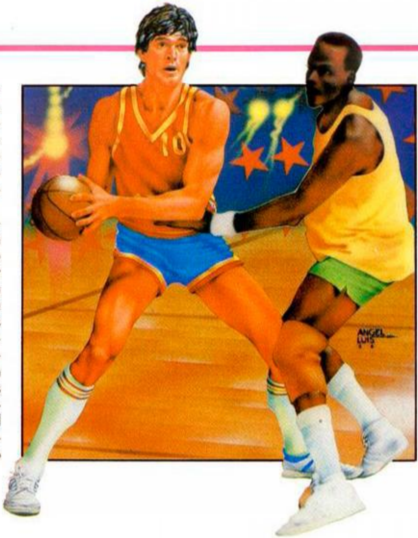
Los primeros pasos del Spectrum por los aros no pudieron ser más desastrosos. **One on one**, de Airolasoft, es uno de esos programas que nunca debieron ser realizados. Comercializado a un precio abusivo, con la engañosa presencia en la carátula de Julius Erving y Larry Bird, el programa se convierte en una pésima versión del original del Commodore donde todo, absolutamente todo, es negativo, desde el parpadeo de los sprites y la sobreimpresión sobre el fondo hasta la lentitud de movimientos, la escasamente lograda perspectiva y la extrema facilidad del juego que no ofrece ningún motivo de superación al jugador. Su único punto reseñable es haber sido el primer programa basado en la modalidad del «uno contra uno», que más tarde será retomada con resultados radicalmente distintos.

• **World series basketball**, de Imagine, es si no recordamos mal el único simulador de baloncesto que ofrece el control sobre un equipo completo de cinco jugadores. Muy adictivo por la rapidez y dificultad que entraña, el juego ofrece un cómodo sistema en el que el jugador controlado por nosotros se muestra de diferente color, permitiéndose todo tipo de pases entre los jugadores del equipo y controlar fácilmente la dirección de la pelota. Permite cambiar el color del campo, el nivel de dificultad y la duración de los partidos, pero semejante rapidez se paga con una pantalla algo fría e impersonal. Un excelente juego que engancha desde el primer momento y no ha perdido validez pese al largo tiempo transcurrido desde su realización.

• **Two on two**, de Activision, es un programa que por sus innovaciones no puede ser fácilmente ignorado por los usuarios. Curiosa simulación en la que cada equipo controla dos jugadores, ofrece la posibilidad de elegir la estrategia a desarrollar en cada jugada, de modo que el segundo integrante del equipo realice sus movimientos y marcajes de una forma determinada. Incorpora una completísima biblioteca de infracciones y el campo ofrece una interesante y amplia perspectiva. Por

desgracia, los aspectos gráficos parecen haber sido relegados a un segundo término en función de la rapidez de movimientos, por lo que los diseños de jugadores, público y escenario dejan bastante que desear.

La estrella de los programas de baloncesto es, sin duda, el recién lanzado **Fernando Martín basket master**, programa de larga y dolorosa gestación que ha llegado por fin a nuestras pantallas, tras largas dificultades que retrasaron su lanzamiento, previsto en un principio para las pasadas navidades. Sobre la filosofía del «uno contra uno», el programa es perfecto en todos los sentidos. El público vociferante, los aros, la cancha, los jugadores diseñados al máximo detalle (el jugador controlado por el ordenador se parece a Fernando Martín) y los perfectos movimientos se complementan con virguerías tales como ocho tipos de mate, tiros de 6,25, personales, tiros libres (todos se sancionan como uno más uno), rebotes, tapones, repetición ampliada y en cámara lenta de todas las acciones de mate y estadísticas detalladas. Un programa llamado a ser un verdadero éxito que cuenta con una velocidad francamente notable teniendo en cuenta la riqueza gráfica y el tamaño de los jugadores en pantalla.



VOLEYBALL

Solamente un programa de este género ha llegado a nuestros monitores. Estamos hablando del **Bump, set, spike!** de la casa inglesa Mastertronic, los creadores del software barato. Pese al reducido precio del programa (dos libras en Inglaterra, 700 pesetas en España), el juego tiene una calidad razonable amparada en su indiscutible originalidad. Dos jugadores por equipo, mucho más fáciles de manejar que cinco, disputan la victoria en un estadio en estudiada perspectiva isométrica. Gráficos muy simples dan paso a un movimiento ágil y una cuidada sensación de tridimensionalidad.

GOLF

El golf fue curiosamente el deporte sobre el que aparecieron más versiones en los primeros meses tras el lanzamiento del Spectrum, la mayoría impresentables e invendibles teniendo en cuenta el nivel de calidad que hoy por hoy exigen los compradores de software. Nos centraremos, pues, en cuatro programas de relativamente reciente aparición en los que, por fin, se ha conseguido una simulación razonablemente precisa teniendo en cuenta la alta sofisticación que exige este deporte.

• **Konami's golf**, de Imagine, no destaca en el aspecto gráfico, tanto en el

BOLOS

Aunque sea únicamente por curiosidad no podemos dejar de comentar el **10th frame**, de U.S. Gold, que pasará a la historia del software como el único, hasta el momento, simulador de un partido de bolos.

Cuesta acostumbrarse a la combinación de colores utilizada, pero hay que reconocer la fidelidad y variedad de las opciones disponibles, desde las ya imprescindibles para escoger nivel de juego hasta la posibilidad de combinar hasta ocho jugadores y escoger entre liga o partidas sueltas. Habrá que medir al milímetro la fuerza y dirección de la bola en los niveles altos para demostrar que el juego no es tan fácil como pueda parecer a primera vista.



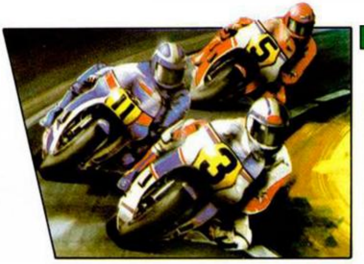
diseño del personaje y el decorado, como en la sensación de perspectiva. Permite escoger entre uno o dos jugadores teniendo en cuenta que ambas posibilidades difieren notablemente, pues con la opción de un jugador nos limitaremos a intentar cubrir decorosamente los nueve hoyos de los que consta el juego (y no 18 como en la realidad), mientras que con la opción de dos jugadores ganará el primero que logre completar cinco hoyos. Como en todos los simuladores de golf existe la imprescindible posibilidad de escoger palo y regular la fuerza y dirección del golpe.

• **Hole in one**, de Mastertronic, te reta a conseguir, como dice su título, un hoyo en un solo golpe. Una vez más el precio no implica calidad, por lo que los que decidan escoger este juego no se verán en absoluto defraudados. Las diferencias con el programa anteriormente comentado radican en la inclusión de marcadores que indican la velocidad del viento y la inclinación del green, a la vez que se mejora notablemente el paisaje y la sensación de realidad derivada de la perspectiva utilizada, que se hace muy cómoda y natural.

• **Nick Faldo plays the open golf** (¡vaya nombrecito!) es el más antiguo de los cuatro juegos comentados, surgiendo en una época donde la utilización de iconos y cursores estaba muy de moda. Inmerso en esta corriente, el programa facilita extraordinariamente la selección y regulación de todos los elementos variables del juego mediante el cómodo empleo de cinco teclas o un joystick para mover la típica flechita y seleccionar las opciones deseadas. Un colorido muy estudiado y una cuidada presentación acompañan a un programa que destaca por la facilidad de su manejo, por lo que atrae rápidamente al usuario, desanimado por la intrínseca complejidad de este deporte. Un simpático caddie nos entrega los palos y nos pregunta si estamos seguros, si la elección no le parece acertada. De todos modos el programa falla en lo más esencial como puede ser el grado de control y respuesta a los controles o presentación visual del campo, por lo que una vez dominado queda muy por debajo del resto de los programas de este grupo.

El mejor programa de golf es, sin duda, el **Leader board**, de U.S. Gold, juego que combina lo mejor de experiencias anteriores para crear una perfecta simulación en la que no falta detalle alguno. Permite escoger nivel de dificultad, número de jugadores y número de hoyos, realiza la selección de movimiento mediante un cómodo sistema de cursores y se acerca extraordinariamente a la realidad al conseguir que cada hoyo tenga unas características definidas de longitud y relieve que hace el recorrido interesante y variado en todo momento. La sensación de perspectiva, aunque mejorable, es la mejor de todos los juegos hasta ahora indicados y solamente podemos achacarle la lentitud en el cambio de pantalla al elaborar los nuevos escenarios con la técnica del fill.





SIMULADORES DE MOTOS

La pasión de la velocidad ha sido una constante en el desarrollo del hombre, el cual, condenado a la lentitud de sus movimientos, ha ideado máquinas cada vez más veloces que le permitan sentir el aire enloquecido sobre la cara y ver el paisaje desvanecerse ante sus ojos. Con una peligrosidad que supera con creces otro tipo de actividades, el conductor de coches o motos que se juega la vida en cada carrera, arriesga su integridad por una afición que, según parece, va más allá de todo lo que los que vamos a pie podemos imaginar.

Muchos y muy buenos simuladores de motos han pasado por nuestras pantallas. El ya mítico **3D Deathchase** era un juego para 16 K (¡qué tiempos aquellos!) en el que nuestra moto se deslizaba evitando chocar contra los árboles en una frenética carrera en persecución de unos fugitivos. **Jump challenge** era un extraño programa en el que, bien en bicicleta, bien en moto, según el nivel, debía-

mos limitarnos a tomar impulso por una rampa para saltar sobre una larga hilera de coches inmóviles. El famoso **Full throttle** es la primera simulación en toda regla de una carrera de motos. Permite escoger el número de vueltas y el circuito en el que correr de entre los varios disponibles, y como en la mayoría de los programas de este tipo salimos en última posición de un nutrido grupo para intentar ir adelantando corredores y acabar la carrera en una posición más digna. Gráficos simples y una inexplicable utilización del color le hacían un programa por encima de la media para los tiempos que corrían, pero inferior a otros simuladores que han ido apareciendo posteriormente.

• **Speed King 2** y **BMX Simulator** son juegos aceptables que no aportan nada nuevo sobre programas más potentes a la vez que demuestran haber sido realizados sin excesivas pretensiones, si bien el Speed King 2 de Mastertronic incluye la curiosa posibilidad de controlar a dos jugadores simultáneos dividiendo la pantalla en dos partes que enfocan a cada uno de los jugadores, desapareciendo el gran grupo de pilotos y quedando reducida la carrera a un interesante mano a mano entre ambos.

• **TT Racer** de Digital integration coloca el listón en una altura considerable, pues refleja como ninguno la sensación que se vive sentados sobre la máquina a esas increíbles velocidades, sensación muy diferente a la que se siente desde las tribunas. De ahí vienen esos extraordinarios efectos de inclinación cuando tomamos una curva o estamos a punto de derrapar. Por lo demás el programa permite escoger entre una amplia gama de cilindradas y circuitos, a la vez que nos obliga a estar pendiente a lo largo de la carrera de los imprescindibles elementos mecánicos tales como velocidad, frenos, combustible y hasta el desgaste de las ruedas.

Todos los halagos posibles, sin embargo, los hemos reservado para un programa de excepcionales cualidades que ha dejado asombrados a todos los que no hemos podido evitar la tentación de cargarlo y comprobar que, efectivamente, todo lo que se había hablado sobre él era cierto. **Enduro racer,** de Activision, figura por derecho propio no solamente entre los mejores simuladores de motos sino entre los mejores programas de todos los tiempos para Spectrum. Todo comentario no hace honor a un programa increíblemente real en el que los excelentes gráficos y la vertiginosa velocidad alcanzada se unen a efectos de inclinación lateral y frontal excepcionalmente reales.

Por si fuera poco, el juego además de bueno es divertido, pues a lo largo de la carrera encontraremos multitud de obstáculos en forma de piedras y hoyos en el camino que, unidos a la inevitable dificultad que supone esquivar al resto de motoristas, hacen que la tensión no disminuya ni un solo instante en nuestra alocada carrera hacia la meta. La posibilidad de realizar hábiles esquives y espectaculares cabriolas ratifican que Enduro racer es, simplemente, un juegazo.

SIMULADORES DE COCHES

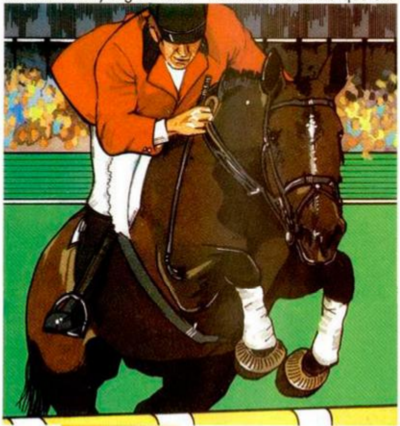
• **Bandera a cuadros,** de Psion, fue el primer simulador de una carrera de Fórmula 1 y consiguió alcanzar un merecido éxito por su indiscutible calidad en una época donde la mayoría de los juegos para Spectrum eran auténticas medianías. Tras escoger un circuito entre los diez disponibles y un coche entre los modelos existentes, la carrera en sí ofrece una correcta sensación de realidad en el movimiento de acercamiento del horizonte y los efectos de vibraciones cuando nuestro coche pincha o atraviesa una zona de piedras, aceite o cristales, lo que, unido a la facilidad de manejo y a la correcta disposición de la pantalla lo convierten en un juego agradable y adictivo.

• **Pole position,** versión de un juego que hizo furor en las máquinas de los bares, surgió como un proyecto bastante ambicioso en el que, sinceramente, todo se quedó en la buena intención. Sorprendente en el rico colorido empleado para la pista y el bólido, el movimiento se revela brusco, pues hace que el paisaje se acerque a saltos, a la vez que el efecto de explosión cuando nuestro coche choca carece de realismo y las maniobras a realizar se efectúan lentamente y con escasa respuesta de los controles. Un juego que ofrecerá limitada diversión a los menos exigentes en un campo donde, a diferencia de otros, no hay una superestrella que podamos recomendar sin temor a equivocarnos.



CABALLOS

Dos curiosos programas con los caballos como protagonistas se han asomado al mercado del software con características y argumentos claramente diferentes. **Grand National,** de Elite, se acerca en cierta medida a los juegos de olimpiadas desde el momento que exige mover a nuestro caballo a base de frenéticos movimientos de dedos, pero incluye una parte de estrategia y suerte al permitir realizar apuestas a cualquiera de los cuarenta caballos participantes, que puede ser el nuestro o no, después de observar la relación de favoritos. El desarrollo de la carrera se asemeja a la realidad, mostrando la pista desde una vista vertical a la vez que una pequeña ventana enfoca a nuestro caballo lateralmente y nos permite calcular adecuadamente el salto. El programa se hace relativamente fácil de dominar a los pocos intentos, por lo que es posible que acabe pronto en el fondo de un cajón, pero antes de hacerlo seguro que consigue divertir a quien lo practique, pues es realmente adictivo y original.



Show jumping, de Alligata, se traslada al mundo de los concursos de saltos donde un elegante jinete acompaña a su cabalgadura en un recorrido plagado de obstáculos a través de varios circuitos de creciente nivel de dificultad. Nos exigirá gran habilidad a la hora de dar velocidad a nuestro caballo o calcular el salto y la dirección adecuada, pues la dificultad en este caso es sumamente elevada. Contiene toda la reglamentación de este tipo de pruebas (tiempo, número de rehuses), pero los gráficos son notablemente más simples que en el juego anterior. De todos modos son programas lo suficientemente distintos como para hacer difícil cualquier comparación entre ellos.

VELA

Sailing, de Activision, es un curioso programa que mezcla sabiamente la habilidad con la estrategia, pues nos permite no solamente escoger el país al que queremos representar sino también las características de nuestro balandro tales como la longitud del mástil, la superficie de las velas, el material empleado, el espacio libre en cubierta o la longitud del casco. La perspectiva y el desarrollo de la regata en sí son aspectos muy bien conseguidos, por lo que una vez superada la sorpresa inicial Sailing puede convertirse en un juego muy adictivo que tardará quizá tiempo en aburrirnos.

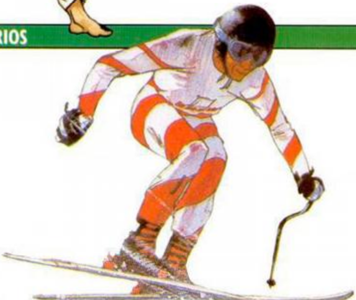




JUDO

Es una verdadera lástima que el único programa sobre tan interesante disciplina sea tan malo de solemnidad. Un deporte bastante adecuado para ser trasladado al ordenador no ha conseguido ser presentado con la calidad que se hubiera merecido. **Uchi mata,** de Martech, es un verdadero bodrio impresentable por su total ausencia de calidad. Si bien la intención parece haber sido buena y se han incorporado todas las llaves más importantes, así como un árbitro y marcadores de la fuerza de ambos contendientes, lo primero que entra por los ojos es el increíble parpadeo de los sprites, lo que unido a la extrema dificultad de conseguir que nuestro judoka obedezca a nuestras órdenes, acaban haciendo de Uchi mata un juego que no se carga dos veces, pues la primera experiencia es generalmente suficientemente frustrante.

VARIOS

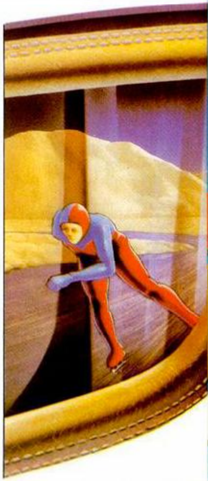


Terminamos este exhaustivo comentario con tres programas de difícil clasificación que, con una estructura muy similar a los juegos de olimpiadas contienen pruebas de lo más variopinto y original.

• **Winter sports,** de Electric dreams, contiene ocho pruebas íntimamente relacionadas con la nieve, de desigual realización gráfica y adicción, por lo que pese al gran interés que pueden despertar pruebas tan interesantes y originales, observamos que la emoción varía demasiado de unas a otras. En cualquier caso, se trata de un programa recomendable por lo novedoso y divertido de su desarrollo, enganchando rápidamente a los furibundos machacadores de teclados. En plan informativo, señalaremos que las ocho pruebas son descenso, slalom, slalom gigante, hockey sobre hielo, salto de esquí, patinaje de velocidad, bobsled y biathlon.

• **Winter games,** de Epyx, tiene un planteamiento prácticamente idéntico al programa de Electric dreams, pero posee unas características gráficas excepcionales en el diseño y colorido de los paisajes escenarios, consiguiendo algunas de las más hermosas pantallas jamás vistas en un Spectrum. Además, la vistosidad, velocidad y adicción de las diferentes pruebas se mantiene con una calidad bastante más alta y uniforme a lo largo de las mismas, por lo que no dudamos en recomendaros este juego integrado por pruebas de salto de esquí, esquí acrobático, patinaje de velocidad, patinaje artístico, bobsled, biathlon y patinaje libre.

• **World games,** de Epyx, es un programa de reciente aparición versión del original, algo más antiguo, para Commodore. Frente a un planteamiento en la más pura línea de los juegos de olimpiadas, World games supone la interesante variación de haber sustituido las pruebas atléticas por eventos de lo más original, cada uno de ellos típicos de un país diferente del mundo. Al igual que el Winter games, permite escoger las pruebas en las que deseamos participar y mantiene una interesante línea de calidad en su diseño gráfico, lo que siempre es de agradecer en una programa que contiene ocho desafíos: levantamiento de pesas en la URSS, salto de barriles en Alemania, salto de la Quebrada en México, slalom en Francia, equilibrio sobre troncos en Canadá, lanzamiento de troncos en Escocia, el típico rodeo de los Estados Unidos y, finalmente, lucha libre japonesa.

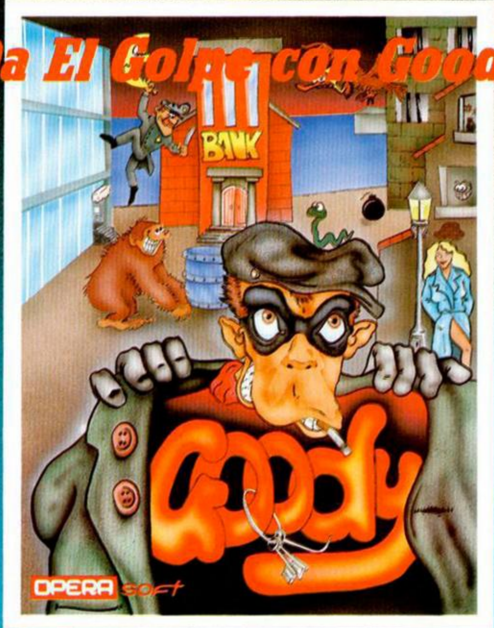


que no digan que no das ni golpe

Goody tiene un buen plan, pero los grandes planes nunca son sencillos, por eso necesita un buen socio como tú, experimentado y audaz.

Esta es tu oportunidad para dar el golpe del Siglo: el asalto al Gran Banco.

Da El Golpe con Goody



Versión para PC y Compatibles

También para Amstrad, MSX, Spectrum, Spectrum + 3 y Commodore

OPERA soft

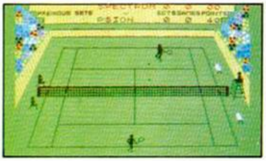
Pza. Santa Catalina de los Donados, 3, 4° Dcha.
28013 Madrid. Tel. 241 92 70 / 241 96 82

Distribuidor en Cataluña
Discovery Informatic
Telfs.: (93) 256 49 08 - 09

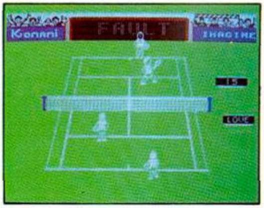
Si no lo encuentras en tu distribuidor habitual llámanos: (91) 241 92 70 - 241 96 82

TENIS

Match Point



Tennis



Ping-Pong



LUCHA

Rocky



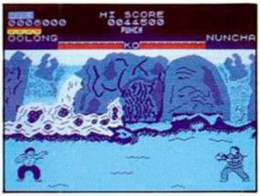
Sai Combat



The way of the tiger



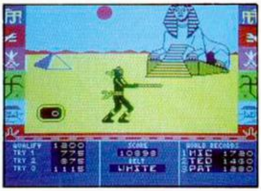
Yie ar Kung-fu



Kung-fu master



Ninja master



BALONCESTO

Fernando Martín



World basketball



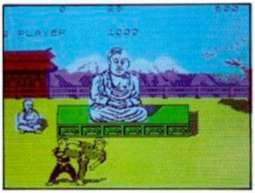
One on One



Internacional kárate



Exploding fist



Shao lin's road



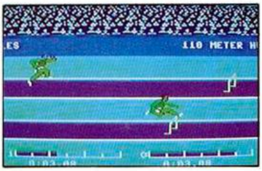
JUDO

Uchi mata

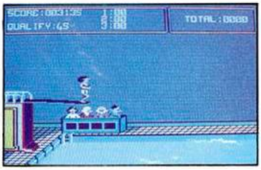


ATLETISMO

Decathlon



Supertest



Hypersport



Videolimpic



FÚTBOL

World-cup



Super Soccer



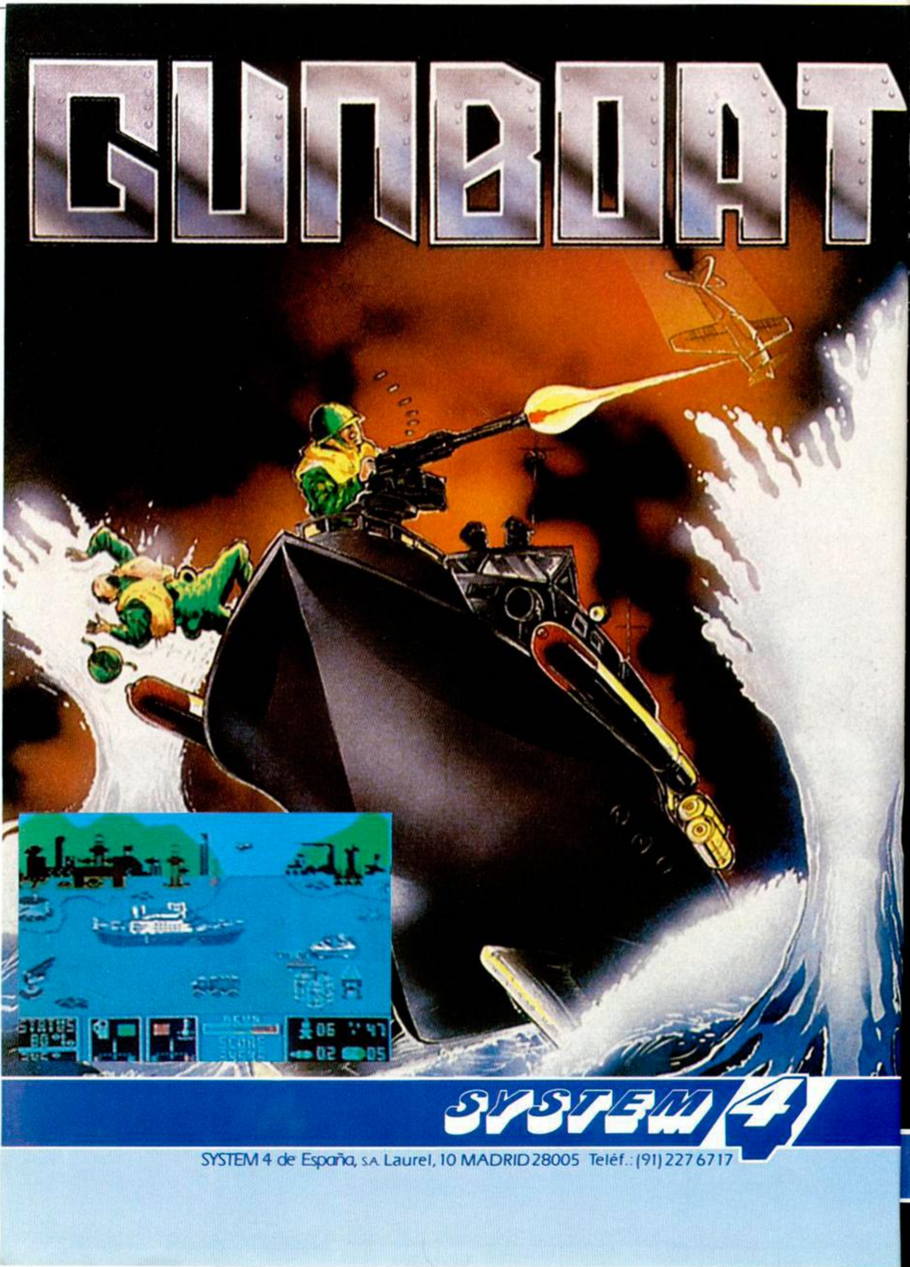
Macth day

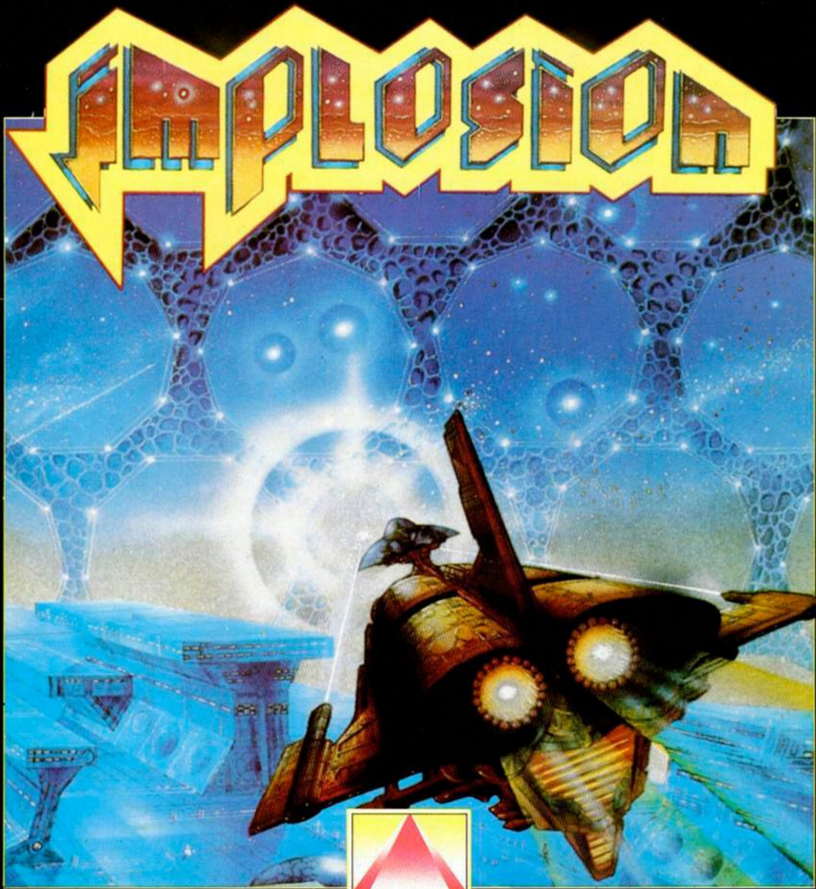


BEISBOL

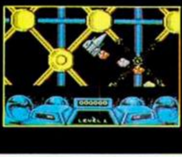
World series baseball



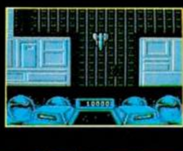

GUNBOAT
SYSTEM 4
SYSTEM 4 de España, S.A. Laurel, 10 MADRID 28005 Teléf.: (91) 227 6717



CASCADE
GAMES LTD







SYSTEM 4

SYSTEM 4 de España, S.A. Laurel, 10 MADRID 28005 Teléf.: (91) 227 6717

Todos, absolutamente todos los grandes títulos del soft, ya sean fantásticas aventuras o incorregibles arcades, protegen de las miradas de los curiosos los secretos que los graduados del joystick deben desvelar. Sólo un pequeño detalle escapa al control de los más osados programadores, nada como una buena pantalla de presentación para que las incógnitas que encierra un juego, dejen de ser tales si hacemos buen uso de nuestra imaginación.

No vamos a ocuparnos esta vez de desvelar misterios, ni de deshacer entuertos galácticos. Nuestro objetivo va mucho más allá. Puestos a investigar hemos optado por rescatar a los olvidados del mundo del software, reuniendo alrededor de una mesa a los artífices de este explosivo invento que se llama pantallas de presentación. Originales diseños que constituyen de por sí auténticas obras de arte, en las que se confunde la concepción clásica del dibujo con las más recientes técnicas de trabajo artístico, pero ante todo, dignas de integrar las filas de cualquier Museo de Arte Contemporáneo con previsión de futuro.

Introducción

Estabamos dispuestos a hacer las cosas bien desde el principio, por eso una vez instalados, cada uno de los asistentes a nuestra improvisada mesa redonda se presentó al resto y a modo de introducción fue comentando brevemente sus aportaciones al mundo del software.

Las presentaciones fueron por «orden de aparición» las siguientes:

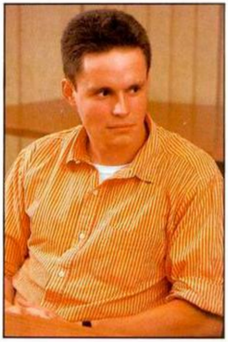
Carlos Granados, componente de Made in Spain, lleva trabajando en el mundo del software desde hace cinco años. Su experiencia, al igual que la de los demás asistentes, no se limita sólo a las pantallas de presentación, sino que además se ocupa del resto de los gráficos. Dentro del tema que nos ocupa obra suya es la pantalla de presentación de «El Misterio del Nilo».

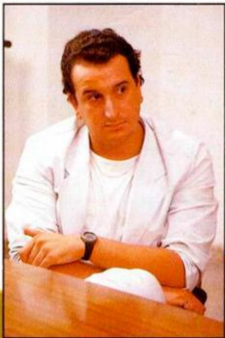
Juan Delcán también trabaja con Made in Spain. Comenzó hace poco más de diez meses y además de la pantalla de presentación del popular «Sir Fred», en la versión de Spectrum, se ocupa del tema gráfico en conjunto de los programas de esta compañía y prepara la pantalla de presentación de un nuevo juego, ambientado en una abadía, que será distribuido por Opera Soft.

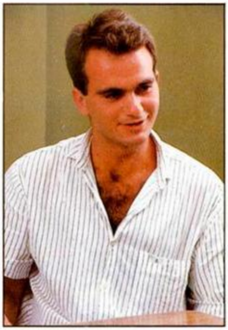
Carlos Díaz miembro fundador de la compañía Opera Soft, se encarga en ella del desarrollo de los gráficos, entre ellos Livingstone Su-

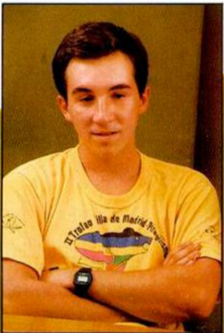
LAS PANTALLAS DE PRESENTACIÓN

DISEÑO GRÁFICO

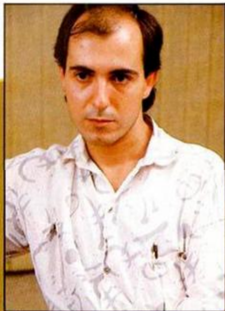


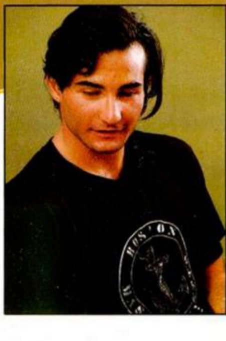












"De izquierda a derecha y de arriba hacia abajo: Gonzalo Suárez, Carlos Díaz, Adolfo López, Carlos Granados, Francisco Javier Mora, Adolfo Fernández, Javier Cano y Juan Delcán. Los nuevos diseñadores del software español"

pongo y el más reciente Last Mission.

Gonzalo Suárez, más conocido como **Gonzo** comenzó a trabajar con Opera Soft hace diez meses. Realizó la pantalla de presentación de Livingstone Supongo y del proyecto más inmediato de Opera Soft Goody, el juego que nos plantea el ambicioso proyecto de robar nada menos que en el Banco de España.

Javier Cano, programador de Topo Soft, ha visto nacer a esta compañía y lleva en el mundo de la informática desde hace tres o cuatro años. Es junto con Carlos Granados el más veterano. Es el autor entre otros gráficos de la carátula del juego Survivor.

Francisco Javier Mora forma parte del equipo de programadores de Dro Soft, se ocupa del aspecto gráfico de los juegos y lleva solamente dos meses y medio programando.

Adolfo Fernández Sánchez comenzó a trabajar con Dro hace aproximadamente año y medio realizando las traducciones de las versiones inglesas de los juegos.

Adolfo López es el único que no se dedica profesionalmente a la programación. Tuvo su primer contacto con el tema gráfico en el primer Concurso de Diseño Gráfico de MICROHOBBY ocupando una de las primeras posiciones con una de sus pantallas y lugares destacados con los demás diseños.

Una cuestión de tiempo

En primer lugar, nos interesaba saber cómo trabajaban cada uno de ellos. Los programas empleados en la confección de la pantalla y el planteamiento general de una carátula eran los temas claves.

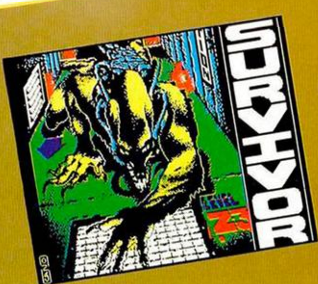
Ante esta pregunta no hubo discursión posible, todos afirmaron realizar primero un boceto en papel que les permitiera observar la perspectiva o la situación de cada elemento y posteriormente pasarlo al ordenador, pero en torno a la utilidad de este boceto hubo sus discrepancias. Desde Juan Delcan que afirma que cuanto mejor sea el boceto, teniendo siempre en cuenta al realizarlo las características del ordenador al que va dirigida la pantalla, mejor sale siempre la carátula, a los más ilusos como Gonzo que pese a reconocer que lo utiliza, afirma que no sirve para nada pues una vez frente a la pantalla la imaginación puede jugar malas pasadas y lo que estaba claro en el papel resulta algo completamente diferente.

Hubo también unanimidad al afirmar que es imposible encontrar un programa de dibujo que se adapte perfectamente a las necesidades de cada persona además de la incomodidad de manejo de la mayoría de los títulos disponibles en el mercado y del intento de abordar demasiados aspectos lo que hace que no sea bueno ninguno de ellos, pero que aun así a pesar de que es mucho mejor crear un programa propio como el desarrollado por Dro Soft, la compañía Melbourne House con el clásico Melbourne Draw alcanza el mayor nivel de aceptación entre ellos aunque siempre con alguna pequeña modificación. También Juan reconoció la utilidad de Art Studio al incluir la posibilidad de trabajar simultáneamente con varias pantallas superpuestas.

El tiempo medio empleado en la realización de una pantalla de presentación fue otra de las cuestiones abordadas. Las respuestas para todos los gustos. Javier Cano nos contó que generalmente se puede tardar uno o dos días en conseguir una pantalla pero que el proceso termina dos semanas más tarde ya que se dan pequeños re-



"Una buena pantalla de presentación predispone a jugar"



Nintendo. Más que un videojuego.

ENTRA EN ACCION

Lánzate a disfrutar del mayor avance en videojuegos: el Sistema de Entretenimiento Nintendo.



Juegos de gran emoción controlados por dos microchips que permiten disfrutar del sistema a dos personas simultáneamente. Deportes, acción y series programables. Una gran variedad de opciones de diversión en constante desarrollo.

Ven a El Corte Inglés y descubre el nuevo Sistema de Entretenimiento Nintendo. Toma el mando y... entra en acción.

El Corte Inglés

Nintendo®

toques hasta que queda perfecta. Gonzo, por el contrario, tarda algo más de tres días en encontrar una idea buena para la carátula y una tarde en preparar ésta. Los componentes de Made in Spain aclararon que el tiempo medio puede variar según el ordenador y según el tiempo disponible, siendo más o menos de una semana entre bocetos y desarrollo total. Como nota cuirosa Adolfo López señaló que tardó dos meses en confeccionar pixel a pixel las pantallas que envió al concurso de diseño, aunque realmente tampoco le dedicaba demasiado tiempo al día; era un entretenimiento y no una necesidad profesional.

Francisco Javier también reseñó en este punto que las diferencias del papel al diseño en ordenador son infinitas y por eso aunque la realización técnica en sí no lleva mucho tiempo sin embargo la elección de una idea adecuada para el ordenador puede suponer muchas horas de investigación.

Manos a la obra

Intentando profundizar en el tema nos surge una duda, ¿qué resulta más difícil a la hora de diseñar una pantalla de presentación? Las opiniones como podéis imaginar fueron para todos los gustos. Juan nos contaba que a él lo que más difícil le resulta es encajar en la pantalla el título y el logo. Charlie por el contrario piensa que lo más difícil es encontrar una buena idea base, ya que esta es la clave del éxito. Sin embargo, el más categórico fue Javier quien afirmó: «Lo más difícil es que guste a todo el mundo».

Javier sentó cátedra; entonces decidimos abordar otras cuestiones. Nos interesaba saber, teniendo en cuenta que todos se ocupaban del tema gráfico en general, cuáles eran las diferencias entre la creación de los gráficos del interior de un programa y la pantalla de presentación.

De sus opiniones pudimos extraer dos ideas a modo de resumen. Por una parte que los gráficos del interior son más trabajosos, porque implican una coherencia y el estudio de cada pantalla, conjuntando el tema estético con la animación de esos gráficos y por otro lado que la pantalla de presentación es más dibujo y hay mucha más libertad y es imprescindible tener conocimientos de dibujo, lo cual fue confirmado por todos los asistentes, ya que quien más quien menos todos tienen o han tenido algún contacto con el campo del diseño, bien con dibujos tipo comic o bien a nivel más profesional como Juan y Gonzalo.

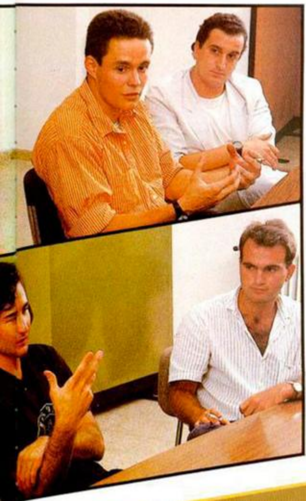
Se comentaron también casos concretos de animación como el popular











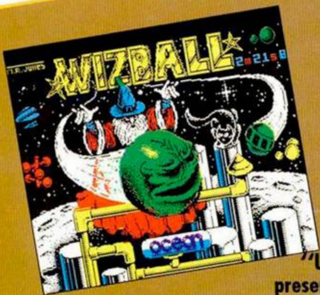
Barbarian en el que para realizar la animación se estudiaron con detenimiento vídeos sobre escenas reales, aunque siempre en el ordenador se deben exagerar, incluso caricaturizar los movimientos para conseguir una animación real, como nos aclaró Juan Delcan.

Hasta aquí todos estaban de acuerdo pero hubo sus más y sus menos al abordar el tema de la mayor o menor libertad al hacer el diseño de la pantalla de presentación. Francisco afirmaba que el dibujo en el ordenador es más frío que ante el papel e incluso a veces puedes tener grandes ideas pero no se pueden aplicar aunque depende de las limitaciones de cada ordenador. Juan recalcó que no puedes plantearte hacer una pantalla igual que un cuadro porque condicionan otros factores, aunque nunca deja de ser arte.

Enlazamos aquí con la utilidad de las pantallas. Todos se mostraron de acuerdo con Javier quien afirmó que la pantalla no tiene, como mucha gente cree, la utilidad de hacernos más agradable la lenta carga de cinta. Es algo para admirar, para que la gente se detenga a observarlo. Incluso como Gonzo especificó que la tendencia actual se dirige a aprovechar las carátulas para ir introduciendo el programa con la presentación de las teclas por ejemplo.

Se resaltó también que todos los elementos de un juego deben ser buenos, por ello la calidad de una pantalla de presentación dice mucho del resto del programa, aunque también hay excepciones y siempre es mucho más alentador hacer una buena pantalla si estamos ante un gran juego. Incluso Juan sentenció: «Cuando una carátula es buena es porque lo ha hecho una







"Una pantalla de presentación debe tener la energía suficiente para comunicar algo, lo que deja el campo abierto a cualquier diseño"

persona que trabaja bien. No creo que un programa pueda ser un buen juego si no tiene una buena pantalla de presentación. Una buena pantalla predispone a jugar».

Todos estaban de acuerdo la utilidad de la pantalla es presentar el juego, una puerta que nos da información a cerca del juego, debe de tener la energía suficiente para comunicar algo, lo que deja el campo abierto a cualquier diseño. Por ello, como afirmó Adolfo Fernández no hay que sacrificar la estética a la verdad, deben ser fiel reflejo del juego. Javier en este momento nos aclaró que una pantalla puede ser más o menos sencilla y puede tener la misma fuerza que una pantalla más recargada, aunque siempre en su caso se intenta acomodar la pantalla a la carátula exterior del juego.

Respecto al futuro de las pantallas de presentación, puesto en duda al popularizarse la carga desde disco, nos permitió comprobar cómo todos dejaron sentada la utilidad de las pantallas y su incredulidad ante una posible desaparición.

Pantallas que pasarán a la historia

No podía faltar en nuestra informal mesa redonda la lista de las mejores pantallas de presentación, teniendo en cuenta que estábamos con unos expertos en la materia. La unanimidad fue total: Dinamic ocupa los primeros lugares de calidad.

Las razones de su efectividad fueron analizadas y ante todo se destacó la figura de Azpiri, el sensacional fichaje de Dinamic para el diseño de sus carátulas exteriores. Sorprendiendo cómo señaló Carlos la similitud de la pantalla con la carátula. Dustin, Army Moves y Camelot Warriors fueron los diseños más mencionados. También se reseñó que Dinamic ha cuidado mucho el tema gráfico y cuentan con un gran equipo.

Charlie de Made in Spain resaltó la gran evolución en este tema, ya que comentó como los primeros juegos de Ultimate cómo Atic Atac o Psst, despertaron su admiración entonces, mientras que ahora quedarían relegados a un justo término medio.

Una invitación

La tarde había sido fructífera pasamos más de dos horas hablando. Temas como la posibilidad que brindaban los concursos para iniciarse en el mundo gráfico habían sido abordados, también proyectamos utopías como una previsible exposición de pantallas de ordenador pero ante todo pudimos obtener una invitación para que todos aquellos interesados en el tema se pongan en contacto con las compañías españolas para presentar sus pantallas y una conclusión muy clara: la importancia de las pantallas de presentación es evidente, bien concebidas como una nueva forma de arte, bien como un elemento más que contribuye al éxito de un juego, permanecerán fieles en su lugar durante años como la antesala a los misterios que ocultan unos cuantos K.







''Dinamic ha cuidado siempre el tema gráfico y cuenta con un gran equipo''

DISEÑO DINAMIC

Dinamic ocupa por derecho propio, desde hace ya algún tiempo un destacado lugar en el panorama del software tanto nacional como internacional. Desde siempre todos sus juegos han estado marcados por un sello indiscutible, un toque personal que los identifica como algo propio y exclusivo de Dinamic.

Quien haya seguido de cerca los progresos de esta popular compañía, sin duda, habrá observado que si hay una constante en todos sus títulos es, sin duda, el cuidado tratamiento gráfico, la continua preocupación por evolucionar.

Prueba del interés de Dinamic en este tema es que curiosamente uno de los primeros programas que aparecieron bajo este sello fue Artist, un diseñador gráfico con el que trabajaron hasta que fue desplazado por el popular Melbourne Draw.

No hace falta afirmar que Dinamic es una de las compañías que más importancia ha dado siempre a las pantallas de presentación y, aunque mucho ha llovido desde Babaliba, Saimazon y Videolimpics, la preocupación por el tema ha sido constante.

Tras estos primeros intentos la sorpresa saltó con Profanation. Obra de Santiago Mora, consiguió una pantalla muy espectacular utilizando sólo dos colores; en esta ocasión se sacrificó el colorido para conseguir una mayor resolución, pero los resultados saltan a la vista. Después llegó la saga Phantomas, también de la mano creativa de Santiago, y por añadidura con el fichaje, para la carátula exterior del juego de Azpiri.

A partir de este momento Dinamic fue consciente de la importancia de un buen dibujante. Sus carátulas comenzaron a acercarse al mundo del comic. Llegó algo más tarde Olé Toro, con el correspondiente escándalo más allá de nuestras fronteras y Camelot Warriors. El autor de las pantallas era en esta ocasión Nacho Ruiz.

Pese a que la calidad de estas pantallas es indiscutible, sin duda han sido sus últimos programas los que han despertado mayor espectación. Dustin, Nonamed y Game Over, programa que sufrió en su propia carátula la implacable mano de la censura británica, consiguieron dejar a más de uno boquiabierto por la similitud con los dibujos exteriores. Los dos primeros fueron obra de Javier Cubedo, y en Game Over las glorias del triunfo se repartieron entre Javier y Nacho.

Tampoco podíamos olvidar a Freddy, aunque comparada con otras pantalla, esta es mucho más sencilla. Javier Cubedo, inspirado en el dibujo de Ventura, otro sensacional dibujante, consigue en Freddy Hardest una pantalla de película que muestra un primer plano del protagonista.

Y con Phantis... llegó la revolución. La exótica protagonista de este juego parece tomar vida en las manos de Javier en la pantalla de presentación y de Azpiri en el dibujo de la carátula. Una pantalla perfecta que es la antesala adecuada para los miles de peligros que se esconden tras ella.

Ésta es a grandes rasgos la muestra clara de la evolución que ha seguido Dinamic. Si hemos llegado hasta aquí, ¿podrá Dinamic aceptar un nuevo récord y superar el listón? Esperemos que dentro de muy poco podamos comprobarlo en un nuevo juego.







LOS VIDEO-JUEGOS SEGA PARA TU CASA SON IGUALES A LOS DE LOS SALONES RECREATIVOS

Son nuevos.

¡Son superdivertidos!

Llenos de color, detalle y sonido.

Los video-juegos Sega son iguales a los que ya conoces de los Salones recreativos pero la consola se conecta a cualquier televisor o monitor que tengas en casa.

La misma calidad de imagen. Resolución gráfica de 256 columnas por 192 líneas. 3 generadores de sonido con 4 octavas y 1 white noise. 64 colores. Movimiento en pantalla: Derecha, izquierda, arriba, abajo, diagonal y parcial. Caracteres 8 × 8 Pixeles, máximo 448, Sprites 8 × 8 Pixeles, máximo 256. Salida de imagen RP o RGB. Cartuchos de 1048 (1 Mega) y Tarjetas de 256 K. Y la consola Master System Sega tiene ROM 128 K, RAM 128 K.

Al comprar la consola Sega te regalarán la tarjeta del juego HANG-ON.

Al comprar la pistola como accesorio te regalarán un cartucho Sega ¡con 3 juegos!



JUEGOS DISPONIBLES

TARJETAS (256 K)
Teddy Boy
Transbot
My Hero
Ghost House
Fighter
Super Tennis
Hang on (de regalo con consola)

CARTUCHOS (1 Mega)
Combo (de regalo con Ligh Phaser)
World Grand Prix
Choplifter
Fantasy Zone
Black Belt
The Ninja
Alex Kidd in the Miracle World
Wonder Boy
Action Fighter

PROXIMAMENTE

TARJETAS (256 K)
Spy us Spy
Bank Panic
Woody Pop

Secret Command
Pro wrestling
Shooting Gallery
Great Golf
Great Ice Hockey
Quartet
Astro Warrion/Pit Pot
Enduro Racer
Missile Defense 3D

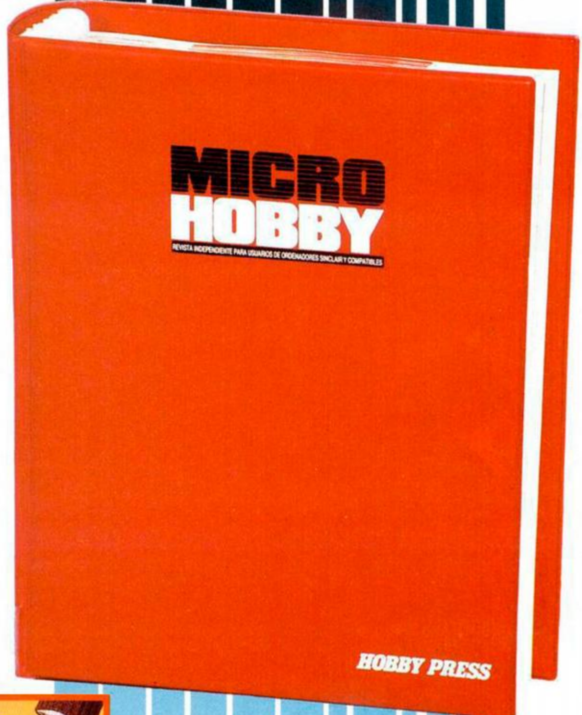
Los video-juegos Sega, así como la consola y la pistola, los encontrarás en tus tiendas habituales de informática, de sonido, o en bazares y grandes almacenes.

Si no encuentras los productos Sega en tu proveedor habitual pide información a:

PRO-IN ELECTRONIC
Velázquez, 10
Tel. 276 22 08/09 MADRID

¡COLECCIONA MICROHOBBY!

850 ptas.



Para solicitar tus tapas, llámanos al tel. (91) 734 65 00



No necesita encuadernación,

gracias a un sencillo sistema de fijación que permite además extraer cada revista cuantas veces sea necesario.

Pablo Ariza

20 RUTINAS ÚTILES

En esta ocasión no os presentamos un único y extenso programa, sino 20 rutinas más o menos cortas que seguro os serán de gran utilidad por su sencillez de manejo y sus buenos resultados. Además, os mostrarán técnicas que os permitirán crear fácilmente otras rutinas adaptadas a vuestras necesidades.



1

Hemos dividido las 20 rutinas en tres grupos principales: gráficas, aritmetico-lógicas y de interrupciones. Las primeras, a su vez, las dividiremos en rutinas de borrado, rutinas de relleno y rutinas de manejo de ventanas.

Casi todas las rutinas necesitan de unos parámetros para funcionar. Con el fin de hacer más fácil su uso desde el Basic, usaremos en todas una técnica gracias a la cual podremos pasar los parámetros a través de las funciones de usuario, una de la instrucciones menos usadas del Spectrum y que demuestran ser de gran utilidad para el intercambio de datos entre el Basic y el Código Máquina.

Como este método va a ser utilizado en casi todas las rutinas, mejor será explicarlo antes de empezar con éstas, y lo haremos ayudados de un ejemplo:

Supongamos que queremos hacer una rutina que haga un doble PEEK, es decir, que dándole una dirección de memoria, tome su contenido y el de la dirección siguiente para formar un número de 16 bits. A esta rutina debemos pasarle un parámetro: la dirección de memoria deseada y ella, a su vez, debe devolvernos otro parámetro: el valor leído de la dirección dada. Pues bien, para ello teclearíamos una línea que fuera más o menos así:

```
1 DEF FN P(D)=USR X
```

X es la dirección donde hemos ubicado la subrutina. Para calcular ahora, por ejemplo, el contenido de la dirección 23675, haríamos PRINT FN P (23675), y obtendríamos un número comprendido entre Ø y 65535, que, en este caso, correspondería a la dirección de los gráficos definidos por el usuario por estar en 23675 la variable del sistema UDG.

Vamos a ver ahora qué es lo que debe hacer nuestra rutina para poder leer el dato que necesita. Cuando el Basic se encuentra con una función FN busca su DEF FN correspondiente y almacena, en la variable del sistema DEFADD (23563), la dirección de memoria donde se encuentran los parámetros de la función.

En nuestro ejemplo, almacenaría la dirección en la que se encontrara la «D» inmediatamente posterior al paréntesis de apertura. Aun-



que en el listado lo que hay a continuación de la D es el paréntesis de cierre, en la memoria hay seis bytes entre medias: un código 14, indicador de número y cinco bytes que almacenarán el valor que se le ha dado al argumento de la función en el FN (siguiendo con nuestro ejemplo, estos cinco bytes codificarían el número 23675).

Por tanto, si desde nuestra rutina en Código Máquina tomáramos el contenido de la variable DEFADD y lo incrementáramos en dos, obtendríamos la dirección donde se encuentra el valor que necesitamos. Como dicho valor será una dirección de memoria y, por tanto, un entero comprendido entre 0 y 65535, los cinco bytes no estarán codificados en coma flotante, sino como «pequeños enteros». Esto quiere decir, como podemos encontrar en el manual del Spectrum (sólo el del 48 K o el del +2), que los dos primeros bytes y el último serán 0, y el número lo podremos encontrar en los bytes tercero y cuarto, por lo que el procedimiento final para obtener el número sería coger el contenido de DEFFAD, sumarle cuatro y co-

ger el contenido de esta dirección que acabamos de formar.

Devolver el resultado desde el Código Máquina al Basic es mucho más fácil: basta con cargar el número en el registro BC antes de retornar para que se convierta en el resultado de la función USR y, por tanto, en el de la FN.

Existen más complicaciones cuando los valores que debemos tratar no son enteros o debemos manejar parámetros alfanuméricos en lugar de numéricos, pero esto lo iremos viendo en cada caso particular.

RUTINAS GRÁFICAS

Probablemente las rutinas gráficas más sencillas sean las que borran la pantalla de alguna forma especial, así que empezaremos por ellas. Tanto para éstas como para todas las demas, existen dos listados, diferenciados por las letras A y B, que son, respectivamente, el listado hexadecimal y el ensamblador. No hay que teclear ambos listados, sino sólo uno de ellos. El hexadecimal es para aquellos que simplemente quieran usar las rutinas sin más complicaciones, mientras que el ensamblador es para los iniciados en el arte del Código Máquina que deseen adentrarse en el funcionamiento interno de las rutinas y modificarlas a su antojo.

1 BORRADO DE PANTALLA POR DEGRADACIÓN DE COLORES

Longitud: 37 bytes.
Dirección: reubicable (cualquier dirección).
Forma de uso: RANDOMIZE USR dirección de comienzo.

Esta rutina borra la pantalla decrementando los colores hasta el negro, produciendo un curioso efecto de brillo. Para conseguirlo se efectúa ocho veces un bucle en el que se realizan los siguientes pasos:

— Se hace una pequeña pausa con dos HALTs para que el efecto no sea excesivamente rápido.

— Se realiza un bucle que se repetirá 768 veces: una por cada atributo de la pantalla. Para salir del bucle comprobamos si HL (que señala al atributo que vamos a tratar) es mayor o igual a 22528, o lo que es lo mismo, si sigue perteneciendo a una dirección de atributos. En realidad, se comprueba sólo el byte alto, que debe ser 88 o mayor. Para cada byte de atributos se hace lo siguiente:

— Se comprueba si el color de la tinta es 0, y de no serlo, se decrementa.

— Se comprueba si el color del papel es 0, y de no serlo, se decrementa.

— Se actualiza el atributo con los nuevos colores de papel y tinta, y con los mismos niveles de FLASH y BRIGHT.

LISTADO A

Línea	datos	control
1	060821FF5A76767E57E6	1071
2	0728013D5F7AE6382802	654
3	D6083AAE63FAA772B7C	1320
4	FE5830E510DEC9000000	1058

DUMP: 60.000
N.° DE BYTES: 37

LISTADO B

```
10 *C-
20 *D+
30 ;
40 ;    **CLS DE COLORES**
50 ;
60 ;
70 CLSCOL LD   B,8
80 ESPEBO LD   HL,23295
90        HALT
100       HALT
110 BUBO  LD   A,(HL)
120       LD   D,A
130       AND  7
140       JR   Z,YACE
150       DEC  A
160 YACE  LD   E.A
170       LD   A,D
180       AND  56
190       JR   Z,TAMCE
200       SUB  8
210 TAMCE OR   E
220       XOR  D
230       AND  63
240       XOR  D
250       LD   (HL),A
260       DEC  HL
270       LD   A,H
280       CP   88
290       JR   NC,BUBO
300       DJNZ ESPEBO
310       RET
```

2 BORRADO DE PANTALLA POR BANDAS LATERALES

Longitud: 225 bytes.
Dirección: reubicable.
Forma de uso: colocar en alguna línea DEF FN C(N,C)=USR dirección y hacer RANDOMIZE FN C (N,C).

C es un valor entre Ø y 255 que indica los nuevos atributos que queremos y se construye de la forma habitual:
128*FLASH+64*BRIGHT +8*PAPER+INK.

N es un número comprendido entre Ø y 15 que indica porqué lados han de salir las bandas. Sus valores son:

1 Arriba.
2 Abajo.
4 Izquierda.
8 Derecha.

Para que salgan bandas por más de un lado no tenemos más que sumar los números correspondientes. Aquí pasamos los parámetros a la rutina de la forma

explicada anteriormente, sólo que en este caso se trata de dos parámetros. Entre el primer parámetro y el segundo se encuentran en la memoria el código de la coma, el código ASCII del nombre del segundo parámetro y el código 14 indicador de número, además de los cinco bytes del valor del primer parámetro, que hacen ocho en total, con lo que si el primer valor estaba en (DEFADD)+4, el segundo estará en (DEFADD)+12. La primera parte de la rutina, de CLSBAN a CLSCOR, no forma parte propiamente dicha de la rutina, sino que se trata de un reubicador que permite que ejecutemos en cualquier dirección una rutina que tenga CALLs y/o JPs. Con este propósito tenemos al final, en TABLA, una lista de todos los puntos de la rutina en los que se hace referencia a alguna dirección absoluta. El funcionamiento de la rutina también es bastante sencillo. En primer lugar, ayudado por la tabla TALAR, calcula cuánto va a tardar en borrar la pantalla, ya que esto depende de las bandas que queramos que haya. A continuación, realiza un bucle en el que llama a las subrutinas que dibujan las bandas en los lados que hemos indicado. La subrutina UP borra una línea horizontal de 32 caracteres (en realidad, sólo borra los atributos con papel y tinta del mismo color). DOWN es igual que UP, sólo que empieza por abajo. LEFT borra una fila vertical de 24 caracteres, al igual que RIGHT.

Una vez se ha terminado con las bandas, es cuando realmente se borra la pantalla, aunque esta operación le resulta inadvertida al usuario.

LISTADO A

línea	datos	control
1	21BF0009EB1A136F1AFE	904
2	FF2815136709E5DDE11A	1148
3	136F1A136709DD7500DD	846
4	740118E3DD2A0B5CDD7E	1081
5	0CE6F8471F1F1FE607B0	1067
6	32ECEA3202EBDD7E0432	1208
7	AAEA06004F210FEB0946	851
8	0E00763E001FDCD6EA1F	924
9	DCF2EA1FDCF8EA1FDC09	1689
10	EB0C79B838E821004011	954
11	014001001875EDB001FF	876
12	02DD7E0C77EDB0C9F579	1460
13	26168787878787CB1487CB	1161
14	146F545D1CC5011F0036	619
15	00EDB0C1F1C9F53E1791	1523
16	18E069F52658112000 3E	835
17	5B360019BC20FAF1C93E	1144
18	1F916F18EA0118180C20	638
19	18180C2018180C101010	200
20	0C33008C003600A2003C	479
21	004A004D007600510092	496
22	00550098005900A90042	561
23	00AF00FFFF0000000000	685

DUMP: 60.000
N.° DE BYTES: 225

LISTADO B

```
10 *C-
20 *D+
30 ;
40 ;     **CLS EN CORTINA**
50 ;
60 CLSBAN LD   HL,TABLA-CLSBA
N
70        ADD  HL,BC
80        EX   DE,HL
90 BUREUB LD   A,(DE)
100       INC  DE
110       LD   L,A
120       LD   A,(DE)
130       CP   255
140       JR   Z,CLCORT
150       INC  DE
160       LD   H,A
170       ADD  HL,BC
180       PUSH HL
190       POP  IX
200       LD   A,(DE)
210       INC  DE
220       LD   L,A
230       LD   A,(DE)
240       INC  DE
250       LD   H,A
260       ADD  HL,BC
270       LD   (IX+0),L
280       LD   (IX+1),H
290       JR   BUREUB
300 ;
310 CLCORT LD  IX,(23563)
320       LD   A,(IX+12)
330       AND  248
340       LD   B,A
350       RRA
360       RRA
370       RRA
380       AND  7
390       OR   B
400 EAT1  LD   (AT1+1),A
410 EAT2  LD   (AT2+1),A
420       LD   A,(IX+4)
430 EUSO  LD   (USO+1),A
440       LD   B,0
450       LD   C,A
460 ETALAR LD  HL,TALAR
470       ADD  HL,BC
480       LD   B,(HL)
490       LD   C,0
500 BUCLS HALT
510 USO   LD   A,0
520       RRA
530 EUP   CALL C,UP
540       RRA
550 EDOWN CALL C,DOWN
560       RRA
570 ELEFT CALL C,LEFT
580       RRA
590 ERIGHT CALL C,RIGHT
600       INC  C
610       LD   A,C
620       CP   B
630       JR   C,BUCLS
640       LD   HL,16384
650       LD   DE,16385
660       LD   BC,6144
670       LD   (HL),L
680       LDIR
690       LD   BC,767
700       LD   A,(IX+12)
710       LD   (HL),A
720       LDIR
730       RET
740 ;
750 UP    PUSH AF
760       LD   A,C
770 COMUN LD   H,22
780       ADD  A,A
790       ADD  A,A
800       ADD  A,A
810       ADD  A,A
820       RL   H
830       ADD  A,A
840       RL   H
850       LD   L,A
860       LD   D.H
870       LD   E,L
880       INC  E
890       PUSH BC
900       LD   BC,31
910 AT1   LD   (HL),0
920       LDIR
930       POP  BC
940       POP  AF
950       RET
960 ;
970 DOWN  PUSH AF
980       LD   A,23
990       SUB  C
1000      JR   COMUN
1010 ;
1020 LEFT LD   L,C
1030 COLERI PUSH AF
1040      LD   H,88
1050      LD   DE,32
1060      LD   A,91
1070 AT2  LD   (HL).0
1080      ADD  HL,DE
1090      CP   H
1100      JR   NZ,AT2
1110      POP  AF
1120      RET
1130 ;
1140 RIGHT LD  A,31
1150      SUB  C
1160      LD   L.A
1170      JR   COLERI
1180 ;
1190 TALAR DEFB 1,24,24,12
1200      DEFB 32,24,24,12
1210      DEFB 32,24,24,12
1220      DEFB 16,16,16,12
1230 ;
1240 TABLA DEFW EAT1+1-CLSBAN,
AT1+1-CLSBAN
1250      DEFW EAT2+1-CLSBAN,
AT2+1-CLSBAN
1260      DEFW EUSO+1-CLSBAN,
USO+1-CLSBAN
1270      DEFW EUP+1-CLSBAN,U
P-CLSBAN
1280      DEFW EDOWN+1-CLSBAN
,DOWN-CLSBAN
1290      DEFW ELEFT+1-CLSBAN
,LEFT-CLSBAN
1300      DEFW ERIGHT+1-CLSBA
N,RIGHT-CLSBAN
1310      DEFW ETALAR+1-CLSBA
N,TALAR-CLSBAN
1320      DEFW 0FFFF
```

3 RELLENADO DE FIGURAS CON TRAMAS (FILL)

Longitud: 403 bytes.
Dirección: reubicable.
Forma de uso: colocar en alguna línea DEF FN F(X,Y,N)=USR dirección y hacer RANDOMIZE FN F (X,Y,N)

X e Y son las coordenadas en alta resolución de un punto interior de la figura que queremos rellenar. N es un número entre 0 y 21 que hace referencia a uno de los 21 GDUs. El GDU elegido será el que se use como trama. Es necesario advertir que esta rutina usa para su funcionamiento una zona de 6 Kbytes colocada en la dirección 32768, por lo que ni la propia rutina ni ninguna otra cosa puede ser colocada en esta zona.

El algoritmo de la rutina

LISTADO A

línea	datos	control
1	215D0109EB1A136F1AFE	807
2	FF2815136709E5DDE11A	1148
3	136F1A136709DD7500DD	846
4	740118E3DD2A0B5CDD4E	1033
5	04DD460CDD7E1421FFFF	1217
6	F5E5C52100801101 8001	979
7	FF1775EDB0C13EAF90CD	1587
8	602257CD60E6CD85EB7A	1608
9	E60757AF32BBEB32BCEB	1444
10	7BA6207E7B66773EC0AC	1297
11	677BB6777CEEC0677C3C	1368
12	474DE607200A79C6204F	857
13	380478D6084778FE5830	983
14	180AA33E00200F3ABCEB	787
15	A7200C7A0F0F0FB047C5	822
16	3EFF32BCEB4D7C3D47EE	1361
17	C7E607200A79D6204F38	788
18	0478C6084778FE403818	919
19	0AA33E00200F3ABBEBA7	929
20	200C7A0F0F0FB047C53E	717
21	FF32BBEBCB0B7A3CE607	1360
22	5720852C7DE61FC2BAEA	1296
23	E17CBD20033C28137C07	823
24	0707E607577CE61FF640	1033
25	67CDB0EBC3ACEAF18787	1831
26	87ED5B7B5C835F8A9357	1276
27	210080D57CE607835F8A	1099
28	93571A2FA6CBBCCBF4AE	1485
29	77CBFCCBB4D1237CFE9C	1735
30	20E3C97BA6C07BBE381A	1336
31	16001E807DE61FC82D7E	937
32	A728F7CB4728022CC91F	1046
33	D8CB031518F93E089247	1003
34	7E1F10FD18EF7A1E80A7	1136
35	C8CB0B3D20FBC900004A	1033
36	0050014D002501DA005A	504
37	004A005001F2005001F5	723
38	004000F5004C0055005B	573
39	0158005C018A005C01BB	600
40	005B0199005C01CA005B	631
41	01FFFF00000000000000	511

DUMP: 60.000
N.° DE BYTES: 403

es el siguiente:

— Buscamos el primer punto «pintado» que haya a la izquierda del punto dado.

— Pintamos el punto obtenido.

— Comprobamos si los puntos superior e inferior a éste están o no pintados.

— Si no lo están, los almacenamos en la pila para su uso posterior.

— Pasamos al punto de la derecha y si no está pintado repetimos el proceso desde el segundo paso.

— Comprobamos si hay algún punto almacenado en la pila y de ser así, repetimos con él el proceso desde el primer paso.

Hasta aquí el algoritmo para un FILL normal. Para hacer un FILL con tramas debemos usar una especie de duplicado de la pantalla, que es el que tenemos en la dirección 32768. Al principio, limpiaremos este duplicado y después iremos pintando en él todos los puntos que pintemos también en la pantalla. Una vez terminado el rellenado, procedemos a sustituir por el entramado deseado toda la zona de la pantalla que haya quedado pintada en el duplicado.

El funcionamiento interno de la rutina sigue fielmente los pasos del algoritmo que acabamos de explicar. Lo único que puede dificultar un poco su comprensión es que a lo largo de todo el proceso se usan directamente direcciones de pantalla en lugar de coordenadas para conseguir que la rutina sea un poco más rápida.

Las coordenadas se dan igual que en Basic, sólo que la Y puede tomar también valores entre —15 y —1 para puntos de las dos líneas inferiores de la pantalla.

LISTADO B

```
10 *C-
20 *D+
30 ;
40 ;        **FILL CON TRAMAS**
50 ;
60 FILL    LD   HL,TABLA-FILL
70         ADD  HL,BC
80         EX   DE,HL
90 BUREUB  LD   A,(DE)
100        INC  DE
110        LD   L,A
120        LD   A,(DE)
130        CP   255
140        JR   Z,FINREU
150        INC  DE
160        LD   H,A
170        ADD  HL,BC
180        PUSH HL
190        POP  IX
200        LD   A,(DE)
210        INC  DE
220        LD   L,A
230        LD   A,(DE)
240        INC  DE
250        LD   H,A
260        ADD  HL,BC
270        LD   (IX+0),L
280        LD   (IX+1),H
290        JR   BUREUB
300 FINREU LD   IX,(23563)
310        LD   C,(IX+4)
320        LD   B,(IX+12)
330        LD   A,(IX+20)
340 RFILL  LD   HL,65535
350        PUSH AF
360        PUSH HL
370        PUSH BC
380        LD   HL,32768
390        LD   DE,32769
400        LD   BC,6143
410        LD   (HL),L
420        LDIR
430        POP  BC
440        LD   A,175
450        SUB  B
460        CALL #2280
470        LD   D,A
480 EIGP   CALL EIGUD
490 ENTRY  CALL LEFT
500        LD   A,D
510        AND  7
520        LD   D,A
530        XOR  A
540 INFLU  LD   (FLUP),A
550 INFLD  LD   (FLDO),A
560 BUINT  LD   A,E
570        AND  (HL)
580        JR   NZ,EOL
590        LD   A,E
600        OR   (HL)
610        LD   (HL),A
620        LD   A,192
630        XOR  H
640        LD   H,A
650        LD   A,E
660        OR   (HL)
670        LD   (HL),A
680        LD   A,H
690        XOR  192
700        LD   H,A
710        LD   A,H
720        INC  A
730        LD   B,A
740        LD   C,L
750        AND  7
760        JR   NZ,DOCAL
770        LD   A,C
780        ADD  A,32
790        LD   C,A
800        JR   C,DOCAL
810        LD   A,B
820        SUB  8
830        LD   B,A
840 DOCAL  LD   A,B
850        CP   88
860        JR   NC,IMPD
870        LD   A,(BC)
880        AND  E
890        LD   A,0
900        JR   NZ,ALZADO
910 COFLD  LD   A,(FLDO)
920        AND  A
930        JR   NZ,IMPD
940        LD   A,D
950        RRCA
960        RRCA
970        RRCA
980        OR   B
990        LD   B,A
1000       PUSH BC
1010       LD   A,255
1020 ALZADO LD  (FLDO),A
1030 IMPD  LD   C,L
1040       LD   A,H
1050       DEC  A
1060       LD   B,A
1070       XOR  7
1080       AND  7
1090       JR   NZ,UPCAL
1100       LD   A,C
1110       SUB  32
1120       LD   C,A
1130       JR   C,UPCAL
1140       LD   A,B
1150       ADD  A,8
1160       LD   B,A
1170 UPCAL LD   A,B
1180       CP   64
1190       JR   C,FINPO
1200       LD   A,(BC)
1210       AND  E
1220       LD   A,0
1230       JR   NZ,ALZAUP
1240 COFLU LD   A,(FLUP)
1250       AND  A
1260       JR   NZ,FINPO
1270       LD   A,D
1280       RRCA
1290       RRCA
1300       RRCA
1310       OR   B
1320       LD   B,A
1330       PUSH BC
1340       LD   A,255
1350 ALZAUP LD  (FLUP),A
1360 FINPO RRC  E
1370       LD   A,D
1380       INC  A
1390       AND  7
1400       LD   D,A
1410       JR   NZ,BUINT
1420       INC  L
1430       LD   A,L
1440       AND  31
1450 BUIP  JP   NZ,BUINT
1460 EOL   POP  HL
1470       LD   A,H
1480       CP   L
1490       JR   NZ,POSIB
1500       INC  A
1510       JR   Z,TRAMA
1520 POSIB LD   A,H
1530       RLCA
1540       RLCA
1550       RLCA
1560       AND  7
1570       LD   D,A
1580       LD   A,H
1590       AND  31
1600       OR   64
1610       LD   H,A
1620 EIGP2 CALL EIGUD
1630 ENTRPO JP  ENTRY
1640 ;
1650 TRAMA POP  AF
1660       ADD  A,A
1670       ADD  A,A
1680       ADD  A,A
1690       LD   DE,(23675)
1700       ADD  A,E
1710       LD   E,A
1720       ADC  A,D
1730       SUB  E
1740       LD   D,A
1750       LD   HL,32768
1760 BUTRAM PUSH DE
1770       LD   A,H
1780       AND  7
1790       ADD  A,E
1800       LD   E,A
1810       ADC  A,D
1820       SUB  E
1830       LD   D,A
1840       LD   A,(DE)
1850       CPL
1860       AND  (HL)
1870       RES  7,H
1880       SET  6,H
1890       XOR  (HL)
1900       LD   (HL),A
1910       SET  7,H
1920       RES  6,H
1930       POP  DE
1940       INC  HL
1950       LD   A,H
1960       CP   156
1970       JR   NZ,BUTRAM
1980       RET
1990 ;
2000 LEFT  LD   A,E
2010       AND  (HL)
2020       RET  NZ
2030       LD   A,E
2040       CP   (HL)
2050       JR   C,INBYTE
2060       LD   D,0
2070       LD   E,128
2080 BULEF LD   A,L
2090       AND  31
2100       RET  Z
2110       DEC  L
2120       LD   A,(HL)
2130       AND  A
2140       JR   Z,BULEF
2150       BIT  0,A
2160       JR   Z,BUC
2170       INC  L
2180       RET
2190 BUC   RRA
2200       RET  C
2210       RLC  E
2220       DEC  D
2230       JR   BUC
2240 INBYTE LD  A,8
2250       SUB  D
2260       LD   B,A
2270       LD   A,(HL)
2280 BUAJ  RRA
2290       DJNZ BUAJ
2300       JR   BUC
2310 ;
2320 EIGUD LD   A,D
2330       LD   E,128
2340       AND  A
2350       RET  Z
2360 BUEIG RRC  E
2370       DEC  A
2380       JR   NZ,BUEIG
2390       RET
2400 ;
2410 FLUP  DEFB 0
2420 FLDO  DEFB 0
2430 ;
2440 TABLA DEFW EIGP+1-FILL,EI
GUD-FILL
2450       DEFW ENTRY+1-FILL,L
EFT-FILL
2460       DEFW BUIP+1-FILL,BU
INT-FILL
2470       DEFW EIGP+1-FILL,EI
GUD-FILL
2480       DEFW EIGP2+1-FILL,E
IGUD-FILL
2490       DEFW ENTRPO+1-FILL,
ENTRY-FILL
2500       DEFW ENTRPO+1-FILL,
ENTRY-FILL
2510       DEFW INFLU+1-FILL,F
LUP-FILL
2520       DEFW INFLD+1-FILL,F
LDO-FILL
2530       DEFW COFLD+1-FILL,F
LDO-FILL
2540       DEFW COFLU+1-FILL,F
LUP-FILL
2550       DEFW ALZADO+1-FILL,
FLDO-FILL
2560       DEFW ALZAUP+1-FILL,
FLUP-FILL
2570       DEFW #FFFF
```

4 "LAVADO" DE UNA FIGURA RELLENADA

Longitud: 53 bytes.
Dirección: reubicable.
Forma de uso: colocar en alguna línea DEF FN W(N)=USR dirección y hacer RANDOMIZE FN W(N)

Esta rutina hace un AND entre la última figura que hayamos rellenado con la anterior rutina y el GDU que especifiquemos. Si hacemos esta operación con un GDU en blanco, anularemos el rellenado. La rutina funciona de forma bastante semejante a la parte que se encargaba de dibujar la trama en la rutina anterior.

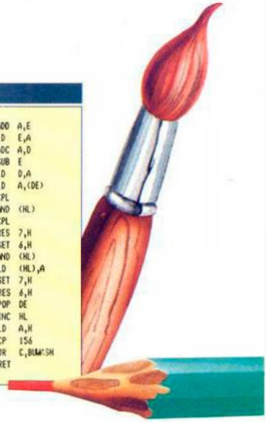
LISTADO A

línea	datos	control
1	DD2A0B5CDD7E04878787	1122
2	ED5B7B5C835F8A935721	1174
3	0080D57CE607835F8A93	1213
4	571A2FA62FCBBCCBF4A6	1377
5	77CBFCCBB4D1237CFE9C	1735
6	38E2C900000000000000	483

DUMP: 60.000
N.° DE BYTES: 53

LISTADO B

```
10 *C-
20 *D+
30 ;
40 ;        ** WASH **
50 ;
60 ;
70          LD   IX,(23563)
80          LD   A,(IX+4)
90 WASH     ADD  A,A
100         ADD  A,A
110         ADD  A,A
120         LD   DE,(23675)
130         ADD  A,E
140         LD   E,A
150         ADC  A,D
160         SUB  E
170         LD   D,A
180         LD   HL,32768
190 BUWASH  PUSH DE
200         LD   A,H
210         AND  7
220         ADD  A,E
230         LD   E,A
240         ADC  A,D
250         SUB  E
260         LD   D,A
270         LD   A,(DE)
280         CPL
290         AND  (HL)
300         CPL
310         RES  7,H
320         SET  6,H
330         AND  (HL)
340         LD   (HL),A
350         SET  7,H
360         RES  6,H
370         POP  DE
380         INC  HL
390         LD   A,H
400         CP   156
410         JR   C,BUWASH
420         RET
```



5 VACIADO DE FIGURAS (UNFILL)

Longitud: 345 bytes.
Dirección: reubicable.
Forma de uso: colocar en alguna línea DEF FN U(X,Y)=USR dirección y hacer RANDOMIZE FN U(X,Y).

Esta rutina es prácticamente idéntica a la de FILL, sólo que vacía zonas rellenas en lugar de rellenar zonas vacías. Al no usar tramas, no necesita la zona de 6 Kbytes que usaban las dos rutinas anteriores.

Pasamos ahora al último apartado de las rutinas gráficas: el de manejo de ventanas.

DUMP: 60.000
N.° DE BYTES: 345

LISTADO A

línea	datos	control	línea	datos	control
1	21230109EB1A136F1AFE	749	19	0FB047C53EFF3281EBCB	1393
2	FF2815136709E5DDE11A	1148	20	0B7A3CE6075720902C7D	862
3	136F1A136709DD7500DD	846	21	E61FC2B6EAE17CBD2002	1443
4	740118E3DD2A0B5CDD4E	1033	22	3CC87C070707E607577C	853
5	04DD460C21FFFFE5C521	1309	23	E61FF64067CD76EBC3A8	1595
6	008011018001FF1775ED	907	24	EA7BA6C87E2FBB301C16	1181
7	B0C13EAF90CDB02257CD	1457	25	001E807DE61FC82D7E3C	975
8	76EBCD47EB7AE60757AF	1485	26	28F73D2FCB4728022CC9	956
9	3281EB3282EB7BA62873	1273	27	1FD8CB031518F93E0892	963
10	7BAE777C3C474DE60720	1017	28	477E2F1F10FD18EE7A1E	958
11	0A79C6204F380478D608	842	29	80A7C8CB0B3D20F8C900	1254
12	4778FE5830180AA33E00	840	30	004600160149 00E700CB	600
13	280F3A82EBA7200C7A0F	826	31	0056004600160 1E20016	427
14	0F0FB047C53EFF3282EB	1206	32	01E5004800E500480051	684
15	4D7C3D47EE07E607200A	857	33	00210154002201780022	310
16	79D6204F380478C60847	903	34	01AC0021018A002201BB	567
17	78FE4038180AA33E0028	793	35	002101FFFF0000000000	544
18	0F3A81EBA7200C7A0F0F	800			

LISTADO B

```
10 *C-
20 *D+
30 ;
40 ;        ** UNFILL **
50 ;
60 ;
70 UNFILL LD   HL,TABLA-UNFIL
L
80        ADD  HL,BC
90        EX   DE,HL
100 BUREUB LD  A,(DE)
110       INC  DE
120       LD   L,A
130       LD   A,(DE)
140       CP   255
150       JR   Z,FINREU
160       INC  DE
170       LD   H,A
180       ADD  HL,BC
190       PUSH HL
200       POP  IX
210       LD   A,(DE)
220       INC  DE
230       LD   L,A
240       LD   A,(DE)
250       INC  DE
260       LD   H,A
270       ADD  HL,BC
280       LD   (IX+0),L
290       LD   (IX+1),H
300       JR   BUREUB
310 FINREU LD  IX,(23563)
320       LD   C,(IX+4)
330       LD   B,(IX+12)
340 FILL  LD   HL,65535
350       PUSH HL
360       PUSH BC
370       LD   HL,32768
380       LD   DE,32769
390       LD   BC,6143
400       LD   (HL),L
410       LDIR
420       POP  BC
430       LD   A,175
440       SUB  B
450       CALL #22B0
460       LD   D,A
470 EIGP  CALL EIGUD
480 ENTRY CALL LEFT
490       LD   A,D
500       AND  7
510       LD   D,A
520       XOR  A
530 INFLU LD   (FLUP),A
540 INFLD LD   (FLDO),A
550 BUINT LD   A,E
560       AND  (HL)
570       JR   Z,EOL
580       LD   A,E
590       XOR  (HL)
600       LD   (HL),A
610       LD   A,H
620       INC  A
630       LD   B,A
640       LD   C,L
650       AND  7
660       JR   NZ,DOCAL
670       LD   A,C
680       ADD  A,32
690       LD   C,A
700       JR   C,DOCAL
710       LD   A,B
720       SUB  8
730       LD   B,A
740 DOCAL LD   A,B
750       CP   88
760       JR   NC,IMPD
770       LD   A,(BC)
780       AND  E
790       LD   A,0
800       JR   Z,ALZADO
810 COFLD LD   A,(FLDO)
820       AND  A
830       JR   NZ,IMPD
840       LD   A,D
850       RRCA
860       RRCA
870       RRCA
880       OR   B
890       LD   B,A
900       PUSH BC
910       LD   A,255
920 ALZADO LD  (FLDO),A
930 IMPD  LD   C,L
940       LD   A,H
950       DEC  A
960       LD   B,A
970       XOR  7
980       AND  7
990       JR   NZ,UPCAL
1000      LD   A,C
1010      SUB  32
```

LISTADO B (sigue)

```
1020        LD   C,A
1030        JR   C,UPCAL
1040        LD   A,B
1050        ADD  A,B
1060        LD   B,A
1070 UPCAL  LD   A,B
1080        CP   64
1090        JR   C,FINPO
1100        LD   A,(BC)
1110        AND  E
1120        LD   A,0
1130        JR   Z,ALZAUP
1140 COFLU  LD   A,(FLUP)
1150        AND  A
1160        JR   NZ,FINPO
1170        LD   A,D
1180        RRCA
1190        RRCA
1200        RRCA
1210        OR   B
1220        LD   B,A
1230        PUSH BC
1240        LD   A,255
1250 ALZAUP LD   (FLUP),A
1260 FINPO  RRC  E
1270        LD   A,D
1280        INC  A
1290        AND  7
1300        LD   D,A
1310        JR   NZ,BUINT
1320        INC  L
1330        LD   A,L
1340        AND  31
1350 BUIP   JP   NZ,BUINT
1360 EOL    POP  HL
1370        LD   A,H
1380        CP   L
1390        JR   NZ,POSIB
1400        INC  A
1410        RET  Z
1420 POSIB  LD   A,H
1430        RLCA
1440        RLCA
1450        RLCA
1460        AND  7
1470        LD   D,A
1480        LD   A,H
1490        AND  31
1500        OR   64
1510        LD   H,A
1520 EIGP2  CALL EIGUD
1530 ENTRPO JP   ENTRY
1540 ;
1550 LEFT   LD   A,E
1560        AND  (HL)
1570        RET  Z
1580        LD   A,(HL)
1590        CPL
1600        CP   E
1610        JR   NC,INBYTE
1620        LD   D,0
1630        LD   E,128
1640 BULEF  LD   A,L
1650        AND  31
1660        RET  Z
1670        DEC  L
1680        LD   A,(HL)
1690        INC  A
1700        JR   Z,BULEF
1710        DEC  A
1720        CPL
1730        BIT  0,A
1740        JR   Z,BUC
1750        INC  L
1760        RET
1770 BUC    RRA
1780        RET  C
1790        RLC  E
1800        DEC  D
1810        JR   BUC
1820 INBYTE LD   A,8
1830        SUB  D
1840        LD   B,A
1850        LD   A,(HL)
1860        CPL
1870 BUAJ   RRA
1880        DJNZ BUAJ
1890        JR   BUC
1900 ;
1910 EIGUD  LD   A,D
1920        LD   E,128
1930        AND  A
1940        RET  Z
1950 BUEIG  RRC  E
1960        DEC  A
1970        JR   NZ,BUEIG
1980        RET
1990 ;
2000 FLUP   DEFB 0
2010 FLDO   DEFB 0
2020 ;
2030 TABLA  DEFW EIGP+1-UNFILL,
EIGUD-UNFILL
2040        DEFW ENTRY+1-UNFILL
,LEFT-UNFILL
2050        DEFW BUIP+1-UNFILL,
BUINT-UNFILL
2060        DEFW EIGP+1-UNFILL,
EIGUD-UNFILL
2070        DEFW EIGP2+1-UNFILL
,EIGUD-UNFILL
2080        DEFW ENTRPO+1-UNFIL
L,ENTRY-UNFILL
2090        DEFW ENTRPO+1-UNFIL
L,ENTRY-UNFILL
2100        DEFW INFLU+1-UNFILL
,FLUP-UNFILL
2110        DEFW INFLD+1-UNFILL
,FLDO-UNFILL
2120        DEFW COFLD+1-UNFILL
,FLDO-UNFILL
2130        DEFW COFLU+1-UNFILL
,FLUP-UNFILL
2140        DEFW ALZADO+1-UNFIL
L,FLDO-UNFILL
2150        DEFW ALZAUP+1-UNFIL
L,FLUP-UNFILL
2160        DEFW 0FFFF
```

6 BORRADO DE UNA VENTANA

Longitud: 114 bytes.
Dirección: reubicable.
Forma de uso: colocar en alguna línea DEF FN B(X,Y,A,L,C)=USR dirección y hacer RANDOMIZE FN B(X,Y,A,L,C).

X e Y son las coordenadas en baja resolución de la esquina superior izquierda de la ventana, A y L son, respectivamente, el ancho y el alto de la misma, y C es el byte de atributos que se construye como ya hemos indicado anteriormente.

LISTADO A

línea	datos	control
1	DD2A0B5CDD7E04570F0F	834
2	0FE6E0DDB60C5F7AE618	1355
3	F64057DD4E1CDD7E1487	1226
4	878747D5626B133600C5	1029
5	0D28040600EDB0C1D114	898
6	7AE607200A3E20835F38	777
7	047AD6085710DEDD7E04	1024
8	161687878787CB1287CB	1143
9	12DDB60C5FDD4614D562	1150
10	6B13C50600DD7E24770D	844
11	2802EDB0C1E111200019	947
12	EB10E7C9000000000000	683

DUMP: 60.000
N.° DE BYTES: 114

LISTADO B

```
10 *C-
20 *D+
30 ;
40 ;**BORRADO DE VENTANAS**
50 ;
60 ;
70 CLWIND LD   IX,(23563)
80        LD   A,(IX+4)
90        LD   D,A
100       RRCA
110       RRCA
120       RRCA
130       AND  224
140       OR   (IX+12)
150       LD   E,A
160       LD   A,D
170       AND  24
180       OR   64
190       LD   D,A
200       LD   C,(IX+28)
210       LD   A,(IX+20)
220       ADD  A,A
230       ADD  A,A
240       ADD  A,A
250       LD   B,A
260 BUBOR PUSH DE
270       LD   H,D
280       LD   L,E
290       INC  DE
300       LD   (HL),0
310       PUSH BC
320       DEC  C
330       JR   Z,NOMAS
340       LD   B,0
350       LDIR
360 NOMAS POP  BC
370       POP  DE
380       INC  D
390       LD   A,D
400       AND  7
410       JR   NZ,INCAR
420       LD   A,32
430       ADD  A,E
440       LD   E,A
450       JR   C,INCAR
460       LD   A,D
470       SUB  8
480       LD   D,A
490 INCAR DJNZ BUBOR
500       LD   A,(IX+4)
510       LD   D,22
520       ADD  A,A
530       ADD  A,A
540       ADD  A,A
550       ADD  A,A
560       RL   D
570       ADD  A,A
580       RL   D
590       OR   (IX+12)
600       LD   E,A
610       LD   B,(IX+20)
620 BBAT  PUSH DE
630       LD   H,D
640       LD   L,E
650       INC  DE
660       PUSH BC
670       LD   B,0
680       LD   A,(IX+36)
690       LD   (HL),A
700       DEC  C
710       JR   Z,NMAT
720       LDIR
730 NMAT  POP  BC
740       POP  HL
750       LD   DE,32
760       ADD  HL,DE
770       EX   DE,HL
780       DJNZ BBAT
790       RET
```

7 INVERSIÓN VERTICAL DE UNA VENTANA

Longitud: 183 bytes.
Dirección: reubicable.
Forma de uso: colocar en alguna línea DEF FN V (X,Y,A,L)= USR dirección y hacer RANDOMIZE FN V(X,Y,A,L)

X,Y,A y L representan lo mismo que en la anterior rutina. En esta ocasión lo que vamos a hacer es volver del revés todo lo que haya en la ventana, es decir, la parte superior pasará a ser la inferior y viceversa. El algoritmo de inversión es el siguiente:

— Calcular la dirección del primer scan de la ventana en DE.

— Calcular la dirección del último byte de la ventana en HL.

— Realizar un bucle N veces, siendo N igual a cuatro por la altura de la ventana, o lo que es lo mismo, la mitad de la altura en pixels. Dentro del bucle hacemos lo siguiente:

—Intercambiar, entre la dirección apuntada por HL

y la apuntada por DE, tantos bytes como el ancho en caracteres de la ventana.

— Calcular el scan siguiente al indicado por DE.

— Calcular el scan anterior al indicado por HL. La estructura de la pantalla del Spectrum es la misma si se la recorre de arriba a abajo que de abajo a arriba, así que si observáis un poco el listado ensamblador podréis comprobar que la forma de calcular el scan siguiente y el anterior tienen una estructura análoga.

El mismo proceso se realiza a continuación para los atributos.

LISTADO B

```
10 *C-
20 *D+
30 ;
40 ; **INVERSION VERTICAL**
50 ;
60 ;
70 VERTIN LD   IX,(23563)
80        LD   A,(IX+4)
90        LD   D,A
100       RRCA
110       RRCA
120       RRCA
130       AND  224
140       OR   (IX+12)
150       LD   E,A
160       LD   A,D
170       AND  24
180       OR   64
190       LD   D,A
200       LD   A,(IX+4)
210       ADD  A,(IX+20)
220       DEC  A
230       LD   H,A
240       RRCA
250       RRCA
260       RRCA
270       AND  224
280       OR   (IX+12)
290       LD   L,A
300       LD   A,H
310       AND  24
320       OR   71
330       LD   H,A
340       LD   B,(IX+28)
350       LD   A,(IX+20)
360       ADD  A,A
370       ADD  A,A
380 BUINV EX   AF,AF'
390       PUSH HL
400       PUSH DE
410       PUSH BC
420 BUCC  LD   C,(HL)
430       LD   A,(DE)
440       LD   (HL),A
450       LD   A,C
460       LD   (DE),A
470       INC  L
480       INC  E
490       DJNZ BUCC
500       POP  BC
510       POP  DE
520       POP  HL
530       INC  D
540       LD   A,D
550       AND  7
560       JR   NZ,BAB
570       LD   A,E
580       ADD  A,32
590       LD   E,A
600       JR   C,BAB
610       LD   A,D
620       SUB  8
630       LD   D,A
640 BAB   DEC  H
650       LD   A,H
660       INC  A
670       AND  7
680       JR   NZ,SUB
690       LD   A,L
700       SUB  32
710       LD   L,A
720       JV   C,SUB
730       LD   A,H
740       ADD  A,8
750       LD   H,A
760 SUB   EX   AF,AF'
770       DEC  A
780       JR   NZ,BUINV
790       LD   A,(IX+4)
800       ADD  A,A
810       ADD  A,A
820       ADD  A,A
830       LD   D,22
840       ADD  A,A
850       RL   D
860       ADD  A,A
870       RL   D
880       OR   (IX+12)
890       LD   E,A
900       LD   A,(IX+4)
910       ADD  A,(IX+20)
920       DEC  A
930       LD   H,22
940       ADD  A,A
950       ADD  A,A
960       ADD  A,A
970       ADD  A,A
980       RL   H
990       ADD  A,A
1000      RL   H
1010      OR   (IX+12)
1020      LD   L,A
1030      LD   B,(IX+20)
1040      LD   A,(IX+20)
1050      SRL  A
1060      RET  Z
1070 BUCT EX   AF,AF'
1080      PUSH BC
1090      PUSH HL
1100      PUSH DE
1110 BUCAT LD  C,(HL)
1120      LD   A,(DE)
1130      LD   (HL),A
1140      LD   A,C
1150      LD   (DE),A
1160      INC  L
1170      INC  E
1180      DJNZ BUCSTR$ T
1190      POP  H\
1200      LD   C,32
1210      ADD  HL,BC
1220      EX   DE,HL
1230      POP  HL
1240      SBC  HL,BC
1250      POP  BC
1260      EX   AF,AF'
1270      DEC  A
1280      JR   NZ,BUAT
1290      RET
```

LISTADO A

línea	datos	control
1	DD2A0B5CDD7E04570F0F	834
2	0FE6E0DDB60C5F7AE618	1355
3	F64057DD7E04DD86143D	1184
4	670F0F0FE6E0DDB60C6F	1128
5	7CE618F64767DD461CDD	1338
6	7E14878708E5D5C54E1A	1167
7	7779122C1C10F7C1D1E1	1220
8	147AE607200A7BC6205F	869
9	38047AD608572257C3CE6	942
10	07200A7DD6206F38047C	715
11	C6086708 3D20CBDD7E04	964
12	8787871616 87CB1287CB	1143
13	12DDB60C5FDD7E04DD86	1234
14	143D2616878787 87CB14	904
15	87CB14DDB60C6FDD461C	1203
16	DD7E14CB3FC808C5E5D5	1480
17	4E1A7779122C1C10F7E1	922
18	0E2009EBE1ED42C1083D	1080
19	20E6C9000000000000000	463

DUMP: 60.000
N.° DE BYTES: 183

INVERSIÓN HORIZONTAL DE UNA VENTANA

Longitud: 172 bytes.
Dirección: reubicable.
Forma de uso: colocar en alguna línea DEF FN H(X,Y,A,L)=USR dirección y hacer RANDOMIZE FN H(X,Y,A,L).

Ahora la inversión la efectuamos en sentido horizontal. En este caso, además de efectuar el intercambio de bytes de forma similar a la inversión vertical, debemos invertir cada byte bit a bit. No importa el orden de estos dos procesos. Nosotros haremos primero el intercambio de bytes. Todo el proceso es el siguiente:

— Efectuar un bucle para cada uno de los scans de la ventana (ocho por el alto en caracteres). Dentro del bucle:

- Calcular en DE la dirección del extremo izquierdo del scan en curso.
- Calcular en HL la dirección del extremo derecho del scan en curso.
- Intercambiar el contenido de la dirección señalada por DE con el de la señalada por HL, incrementar DE y decrementar HL. Este punto se realiza N veces, siendo N el ancho de la ventana en caracteres partido por dos. En el caso de anchos impares despreciamos el resto de la división.

— Ya fuera del bucle, cuando hemos intercambiado todos los bytes de la mitad derecha de la ventana con sus simétricos en la izquierda, realizamos un bucle para todos y cada uno de los bytes que forma parte de la ventana. En este nuevo bucle es donde vamos a invertir los bytes bit a bit. Para ello, hacemos lo siguiente:

- Cargamos en A el byte en cuestión.
- Por ocho veces consecutivas extraemos de A un bit y a través del bit de acarreo lo vamos pasando a E en orden contrario.
- Colocamos el byte así obtenido en la dirección que ocupaba el original.

Como es obvio, para los atributos debemos efectuar el intercambio de bytes, pero no la inversión pixel a pixel. En el listado de la rutina, la inversión de los atributos se hace entre medias de los dos procesos necesarios para la inversión del dibujo propiamente dicho.

LISTADO A

línea	datos	control
1	DD2A0B5CDD7E04570F0F	834
2	0FE6E0DDB60C5F7AE618	1355
3	F64057D5DD461C480DCB	1217
4	38285BDD7E1487878708	967
5	D5627B816FC54E1A7779	1215
6	122D1C10F7C1D1147AE6	1128
7	07200A7BC6205F38047A	679
8	D608570 83D20DADD7E04	979
9	878787161687CB1287CB	1143
10	12DDB60C5FDD7E1408C5	1100
11	627B816FD54E1A777912	1036
12	2D1C10F7E10E2009EBC1	1044
13	083D20E6E1DD461CDD7E	1222
14	148787874FC5E57E1608	1086
15	1FCB131520FA732C10F3	974
16	E1C1247CE607200A7DC6	1180
17	206F38047CD608670D20	697
18	DCC90000000000000000	421

DUMP: 60.000
N.° DE BYTES: 172

LISTADO B

```
10 *C-
20 *D+
30 ;
40 ;**INVERSION HORIZONTAL**
50 ;
60 ;
70 HORINV LD   IX,(23563)
80        LD   A,(IX+4)
90        LD   D,A
100       RRCA
110       RRCA
120       RRCA
130       AND  224
140       OR   (IX+12)
150       LD   E,A
160       LD   A,D
170       AND  24
180       OR   64
190       LD   D,A
200       PUSH DE
210       LD   B,(IX+28)
220       LD   C,B
230       DEC  C
240       SRL  B
250       JR   Z,YAPAS
260       LD   A,(IX+20)
270       ADD  A,A
280       ADD  A,A
290       ADD  A,A
300 BUINV EX   AF,AF'
310       PUSH DE
320       LD   H,D
330       LD   A,E
340       ADD  A,C
350       LD   L,A
360       PUSH BC
370 BUCC  LD   C,(HL)
380       LD   A,(DE)
390       LD   (HL),A
400       LD   A,C
410       LD   (DE),A
420       DEC  L
430       INC  E
440       DJNZ BUCC
450       POP  BC
460       POP  DE
470       INC  D
480       LD   A,D
490       AND  7
500       JR   NZ,BAB
510       LD   A,E
520       ADD  A,32
530       LD   E,A
540       JR   C,BAB
550       LD   A,D
560       SUB  8
570       LD   D,A
580 BAB   EX   AF,AF'
590       DEC  A
600       JR   NZ,BUINV
610       LD   A,(IX+4)
620       ADD  A,A
630       ADD  A,A
640       ADD  A,A
650       LD   D,22
660       ADD  A,A
670       RL   D
680       ADD  A,A
690       RL   D
700       OR   (IX+12)
710       LD   E,A
720       LD   A,(IX+20)
730 BUAT  EX   AF,AF'
740       PUSH BC
750       LD   H,D
760       LD   A,E
770       ADD  A,C
780       LD   L,A
790       PUSH DE
800 BUCAT LD   C,(HL)
810       LD   A,(DE)
820       LD   (HL),A
830       LD   A,C
840       LD   (DE),A
850       DEC  L
860       INC  E
870       DJNZ BUCAT
880       POP  HL
890       LD   C,32
900       ADD  HL,BC
910       EX   DE,HL
920       POP  BC
930       EX   AF,AF'
940       DEC  A
950       JR   NZ,BUAT
960 YAPAS POP  HL
970       LD   B,(IX+28)
980       LD   A,(IX+20)
990       ADD  A,A
1000      ADD  A,A
1010      ADD  A,A
1020      LD   C,A
1030 REINV PUSH BC
1040      PUSH HL
1050 BUBYT LD  A,(HL)
1060      LD   D,8
1070 INBYTE RRA
1080      RL   E
1090      DEC  D
1100      JR   NZ,INBYTE
1110      LD   (HL),E
1120      INC  L
1130      DJNZ BUBYT
1140      POP  HL
1150      POP  BC
1160      INC  H
1170      LD   A,H
1180      AND  7
1190      JR   NZ,BABOOS
1200      LD   A,L
1210      ADD  A,32
1220      LD   L,A
1230      JR   C,BABOOS
1240      LD   A,H
1250      SUB  8
1260      LD   H,A
1270 BABOOS DEC C
1280      JR   NZ,REINV
1290      RET
```

RUTINAS ARITMETICOLÓGICAS

Comenzamos con otro grupo importante de rutinas. La mayoría de ellas usan las facilidades del calculador. El calculador es una de las más potentes utilidades que podemos encontrar en la ROM del Spectrum, y de él se ha hablado más bien poco. Esperamos que con estas rutinas lo conozcáis un poco mejor, aunque para conocerlo más a fondo es recomendable la lectura del libro «The Complete Spectrum Rom Disassembley», de Ian Logan y Frank O'hara, que por desgracia no está traducido al español.

Para acceder al calculador se utiliza la instrucción RST 28, y a partir de entonces, los bytes siguientes son interpretados como códigos que le dicen al calculador las operaciones que deseamos efectuar. Además, existen una serie de subrutinas que sirven para introducir y sacar números de la pila de cálculo, ya que todas las operaciones que podemos efectuar con el calculador, se realizan sobre números almacenados en la pila de cálculo (que no hay que confundir con la pila de máquina direccionada por el registro SP).

9 OPERACIONES LÓGICAS (AND/OR/XOR)

Longitud: 45 bytes.
Dirección: reubicable.
Forma de uso: colocar en alguna línea DEF FN O(A,B,N) = USR dirección, y hacer LET X = FN O(A,B,N).

Las operaciones lógicas AND, OR y XOR, efectuadas bit a bit, son muy sencillas en Código Máquina, pero no en Basic. Gracias a estas rutinas podremos lograrlo fácilmente. A y B son los dos números con los que deseamos efectuar la operación, y deberán ser enteros de 16 bits, es decir, comprendidos entre Ø y 65535. N indica qué tipo de operación es, Ø para AND, 1 para OR y 2 para XOR. El resultado de la operación será el resultado de la función FN. En el ejemplo de forma de uso, el resultado quedaría almacenado en la variable X. En esta rutina aún no nos es necesario utilizar el calculador debido a su enorme sencillez.

DUMP: 6Ø.ØØØ
N.° DE BYTES: 45

LISTADO A

línea	datos	control
1	21220009DD2A0B5CDD7E	789
2	1406B63D280706AE3D28	597
3	0206A678770107000977	549
4	DD7E04DDA60C4FDD7E05	1181
5	DDA60D47C90000000000	672

LISTADO B

```
10 *C-                            100         LD   A,(IX+20)     200         LD   BC,7
20 *D+                            110         LD   B,#B6;OR      210         ADD  HL,BC
30 ;                              120         DEC  A             220         LD   (HL),A
40 ;        **AND/OR/XOR**        130         JR   Z,ID          230         LD   A,(IX+4)
50 ;                              140         LD   B,#AE;XOR     240 ANPO1   AND  (IX+12)
60 ;                              150         DEC  A             250         LD   C,A
70 ANORXO LD    HL,ANPO1+1-ANO    160         JR   Z,ID          260         LD   A,(IX+5)
RXO                               170         LD   B,#A6;AND     270 ANPO2   AND  (IX+13)
80        ADD   HL,BC             180 ID      LD   A,B           280         LD   B,A
90        LD    IX,(23563)        190         LD   (HL),A        290         RET
```



10 LOGARITMO DE CUALQUIER BASE

Longitud: 27 bytes.
Dirección: reubicable.
Forma de uso: colocar en alguna línea DEF FN L(B,A)=USR dirección, y hacer LET X=FN L(B,A).

Con esta rutina calculamos el logaritmo en base B de A. En ésta ya comenzamos a usar el calculador. La fórmula de calcular un logaritmo en base B de A es LN A/LN B. La subrutunia de la ROM STKNUM almacena en la pila de cálculo el número que se encuentre en la dirección señalada por HL. Este número puede ser un «pequeño entero» o una coma flotante. Una vez tenemos los números en la pila, hacemos con ellos las operaciones pertinentes y dejamos el resultado como último valor de la pila. Cuando un programa en Código Máquina llamado con USR retorna al Basic con un RET, el contenido del registro BC se almacena en la pila de cálculo y se toma como el resultado de la función USR. El POP BC del final, extrae de la pila de máquina la dirección de la subrutina que almacena BC, por lo que esta operación no se realiza, y se tomará como resultado de la función USR el número que nuestra rutina ha dejado en la pila de cálculo. De esta forma conseguimos que USR pueda devolver al Basic un valor fuera del margen de enteros entre Ø y 65535.

LISTADO A

línea	datos	control
1	2A0B5C010A0009CDB433	601
2	EF25382A0B5C2323CDB4	932
3	33EF250538C1C9000000	782

DUMP: 60.000
N.° DE BYTES: 27

LISTADO B

```
10 *C-                            63          ADD  HL,BC              140         RST  #28;CALCULADOR
20 *D+                            64          CALL STKNUM             150         DEFB #25;LOG NEP.
30 ;                              70          RST  #28;CALCULADOR     160         DEFB #05;DIVIDIR.
40 ;      **LOGARITMO**           80          DEFB #25;LOG NEP.       170         DEFB #38;FIN
50 ;                              90          DEFB #38;FIN            180         POP  BC
60 ;                              100         LD   HL,(23563)         190         RET
61 LOG     LD   HL,(23563)        110         INC  HL                 200 ;
62         LD   BC,10             120         INC  HL                 210 STKNUM EQU  #3384
                                  130         CALL STKNUM
```

11 RAÍZ ENÉSIMA

Longitud: 27 bytes.
Dirección: reubicable.
Forma de uso: colocar en alguna línea DEF FN R(A,N)=USR dirección, y hacer LET X=FN R(A,N)

A parte de que las operaciones a realizar son distintas, esta rutina es prácticamente igual a la anterior. Realizar la raíz enésima de A es calcular la operación A^(1/N). La toma y devolución de parámetros desde y al Basic se hace de la misma forma que con el logaritmo.

LISTADO A

línea	datos	control
1	2A0B5C2323CDB433EFA1	1051
2	382A0B5C010A0009CDB4	606
3	33EF050638C1C9000000	751



DUMP: 60.000
N.° DE BYTES: 27

LISTADO B

```
10 *C-                               140         LD   HL,(23563)
20 *D+                               150         LD   BC,10
30 ;                                 160         ADD  HL,BC
40 ;    **RAIZ ENESIMA**             170         CALL STKNUM
50 ;                                 180         RST  #28;CALCULADOR
60 ;                                 190         DEFB #05;DIVIDIR
70 RAIZ    LD   HL,(23563)           200         DEFB #06;ELEVAR
80         INC  HL                   210         DEFB #38;FIN
90         INC  HL                   220         POP  BC
100        CALL STKNUM               230         RET
110        RST  #28;CALCULADOR       240 ;
120        DEFB #A1;STK 1            250 STKNUM EQU  #3384
130        DEFB #38;FIN
```

12 GENERACIÓN DE NÚMEROS ALEATORIOS

Longitud: 95 bytes.
Dirección: reubicable.
Forma de uso: colocar en alguna línea DEF FN E(A,B) = USR dirección: DEF FN F(A,B) = USR dirección + 4, y hacer LET X = FN E(A,B) o LET X = FN F(A,B).

En la primera forma de la rutina devuelve como resultado un número entero entre A y B, ambos incluidos. En la segunda devuelve un número racional comprendido entre A y B, incluido A pero no B. Las líneas 340 a 590 del listado ensamblador son una copia exacta de la rutina de la ROM que genera números aleatorios. El resto no son más que operaciones aritméticas para conseguir el resultado deseado. En el primer caso la operación total que se hace es A + INT(RND * (B-A + 1)). En el segundo caso, se hace A + RND * (B-A). Las dos los hace una misma rutina, que se sirve de un indicador para saber si se trata de uno u otro caso. Dicho indicador se inicializa a un valor o a otro según entremos en la rutina por punto o por otro.

LISTADO A

líinea	datos	control
1	3EFF1801AFF52A0B5C23	942
2	23CDB433EF31382A0B5C	960
3	010A0009CDB433EF0103	699
4	38F1F5A72804EFA10F38	1224
5	ED4B765CCD2B2DEFA10F	1230
6	34371604348041000080	506
7	3202A1033138CDA22DED	970
8	43765C7EA72803D61077	962
9	EF040F38F1A72002C1C9	1150
10	EF2738C1C90000000000	728

DUMP: 60.000
N.° DE BYTES: 95

LISTADO B

```
10 #C-
20 #D+
30 ;
40 ; **NUMEROS ALEATORIOS**
50 ;
60 ;
70 INTEG  LD   A,255
80        JR   GUARDA
90 RACIO  XOR  A
100 GUARDA PUSH AF
110 RANDOM LD   HL,(23563)
120        INC  HL
130        INC  HL
140        CALL STKNUM
150        RST  #28;CALCULADOR
160        DEFB #31;DUPLICAR
170        DEFB #38;FIN
180        LD   HL,(23563)
190        LD   BC,10
200        ADD  HL,BC
210        CALL STKNUM
220        RST  #28;CALCULADOR
230        DEFB #01;INTERCAMBIAR
240        DEFB #03;RESTAR
250        DEFB #38;FIN
260        POP  AF
270        PUSH AF
280        AND  A
290        JR   Z,NENTER
300        RST  #28;CALCULADOR
310        DEFB #A1;STK 1
320        DEFB #0F;SUMAR
330        DEFB #38;FIN
340 NENTER LD   BC,(23670)
350        CALL STAKBC
360        RST  #28;CALCULADOR
370        DEFB #A1;STK 1
380        DEFB #0F;SUMAR
390        DEFB #34;STK DATA
400        DEFB #37
410        DEFB #16
420        DEFB #04
430        DEFB #34;STK DATA
440        DEFB #80
450        DEFB #41
460        DEFB #00,#00,#80
470        DEFB #32;MOD
480        DEFB #02;BORRAR
490        DEFB #A1;STK 1
500        DEFB #03;RESTAR
510        DEFB #31;DUPLICAR
520        DEFB #38;FIN
530        CALL FPTOBC
540        LD   (23670),BC
550        LD   A,(HL)
560        AND  A
570        JR   Z,RNDEND
580        SUB  16
590        LD   (HL),A
600 RNDEND RST  #28;CALCULADOR
610        DEFB #04;MULTIPLICAR
620        DEFB #0F;SUMAR
650        DEFB #38;FIN
660        POP  AF
670        AND  A
680        JR   NZ,ENTERO
690        POP  BC
700        RET
710 ENTERO RST  #28;CALCULADOR
720        DEFB #27;INT
730        DEFB #38;FIN
740        POP  BC
750        RET
760 ;
770 STAKBC EQU  #2D2B
780 FPTOBC EQU  #2DA2
790 STKNUM EQU  #33B4
```

13 FACTORIAL

Longitud: 34 bytes.
Dirección: reubicable.
Forma de uso: colocar en alguna línea DEF FN F(N) = USR dirección, y hacer LET X = FN F(N).

La rutina está preparada para el caso especial de 0, cuyo factorial es 1 por convenio. Debido a la forma de almacenar los números en el Spectrum, no se podrá calcular el factorial de números superiores a 32. Si utilizamos números más grandes, o simplemente números no enteros, podremos obtener resultados extraños o bien el error 6 Number too big. El algoritmo de su funcionamiento es el siguiente:

— Introducir un 1 en la pila.
— Tomar el número N.
— Si es 0, retornar, devolviendo 1 como resultado.
— Cargar C con 2.
— Realizar un bucle N-1 veces. De esta forma se sale de la rutina cuando N es 1, cuyo factorial es también 1. Dentro del bucle:

LISTADO A

línea	datos	control
1	EFA138DD2A0B5CDD7E04	1173
2	A728130E024705280DC5	568
3	0600CD2B2DEF0438C10C	803
4	18F0C1C9000000000000	658

DUMP: 60.000
N.° DE BYTES: 34

LISTADO B

```
10 #C-
20 #D+
30 ;
40 ;      **FACTORIAL**
50 ;
60 ;
70 FACT   RST  #28;CALCULADOR
80        DEFB #A1;STK 1
90        DEFB #38;FIN
100       LD   IX,(23563)
110       LD   A,(IX+4)
120       AND  A
130       JR   Z,RESULT
140       LD   C,2
150       LD   B,A
160 BUFAC DEC  B
170       JR   Z,RESULT
180       PUSH BC
190       LD   B,0
200       CALL STKBC
210       RST  #28;CALCULADOR
220       DEFB #04;MULTIPLICAR
230       DEFB #38;FIN
240       POP  BC
250       INC  C
260       JR   BUFAC
270 RESULT POP BC
280       RET
290 ;
300 FPTOA  EQU  #2D05
310 STKBC  EQU  #2D2B
```

- Multiplicar el número de la pila (inicialmente 1) por C.
- Incrementar C.

Para multiplicar C por el contenido de la pila, ponemos B a Ø (por lo que BC será igual a C) y utilizamos la rutina con la ROM STKBC, que almacena en la pila, en formato de cinco bytes, el valor contenido en BC, y después multiplicamos los dos últimos valores contenidos en la pila usando el calculador. Recordemos que cuando hablamos de almacenar números en la pila estamos hablando de la pila de cálculo.

14 CONVERSOR HEXADECIMAL-DECIMAL

Longitud: 41 bytes.
Dirección: reubicable.
Forma de uso: colocar en alguna línea DEF FN H(A$)=USR Dirección, y hacer LET X=FN H(A$).

A$ será una cadena de caracteres que sólo podrá contener números del Ø al 9 y letras mayúsculas de la A a la F. En caso de tener una longitud superior a 4 caracteres, sólo se tomarán como significativos los 4 últimos, pero si tiene más de 255 caracteres, la operación no se efectuará correctamente. Como el resultado será un número comprendido entre Ø y 65535, no necesitamos usar el calculador, sino que lo devolvemos directamente en el registro BC. Esta rutina tiene una complicación que no tenían las anteriores: el parámetro que hay que pasarle no es un número, sino una cadena alfanumérica. En este caso los 5 bytes del parámetro no representarán un número en coma flotante, sino que indicarán la dirección de memoria en la que se encuentra la cadena y su longitud. Además estos 5 bytes no estarán en (DEFADD)+2, sino en (DEFADD)+3, ya que ahora el nombre del parámetro no es una sola letra, sino una letra y el carácter "$". De los cinco bytes, el primero no se usa, los dos siguientes indican la dirección de la cadena y los dos últimos indican su longitud. Si el carácter es una cifra entre Ø y 9, su código estará entre 48-57, y habrá que restarle 48 para obtener su valor hexadecimal. Si es una letra de la A a la F, su código estará entre 65 y 70, y habrá que restarle 55, o 48 y 7, como se hace en la rutina.

LISTADO A

línea	datos	control
1	DD2A0B5CDD5E04DD5605	997
2	DD46062100007 8A72812	675
3	1A13FE3A3802D607D630	898
4	29292929B56F10EEE5C1	1132
5	C9000000000000000000	201

DUMP: 60.000
N.° DE BYTES: 41

LISTADO B

```
10 *C-
20 *D+
30 ;
40 ;**HEXADECIMAL A DECIMAL**
50 ;
60 ;
70 HEXDEC LD   IX,(23563)
80        LD   E,(IX+4)
90        LD   D,(IX+5)
100       LD   B,(IX+6)
110       LD   HL,0
120       LD   A,B
130       AND  A
140       JR   Z,CALCU
150 BUCAL LD   A,(DE)
160       INC  DE
170       CP   58
180       JR   C,NUM
190       SUB  7
200 NUM   SUB  48
210       ADD  HL,HL
220       ADD  HL,HL
230       ADD  HL,HL
240       ADD  HL,HL
250       OR   L
260       LD   L,A
270       DJNZ BUCAL
280 CALCU PUSH HL
290       POP  BC
300       RET
```

15 CONVERSOR DECIMAL-HEXADECIMAL

Longitud: 59 bytes.
Dirección: reubicable.
Forma de uso: colocar en alguna línea DEF FN D$(N)=" "AND USR dirección, y hacer LET X$=FN D$(N).

N debe ser un número entero entre Ø y 65535. La rutina construye una cadena de 4 caracteres que representa el número en hexadecimal. A continuación, gracias a la subrutina STACKSTO de la ROM almacena sus datos (dirección y longitud) en la pila de cálculo, y suma esta cadena con la cadena vacía que se ha almacenado en la pila antes de llamar a la subrutina gracias a la expresión " "AND USR dirección, por lo que al volver al Basic, dicha cadena vacía ya no lo será, sino que se habrá con-

LISTADO A

línea	datos	control
1	213A0009DD2A0B5CDD5E	781
2	04DD56050E0406047BE6	697
3	0FCB1ACB1B10FAFE0A38	1060
4	02C607C630772B0D20E8	892
5	EB010400013CDB22AEF17	946
6	380101000C9000000000	259

DUMP: 60.000
N.° DE BYTES: 59

LISTADO B

```
10 *C-
20 *D+
30 ;
40 ;**DECIMAL A HEXADECIMAL**
50 ;
60 ;
70 DECHEX LD   HL,SPARE+3-DEC
HEX
80        ADD  HL,BC
90        LD   IX,(23563)
100       LD   E,(IX+4)
110       LD   D,(IX+5)
120       LD   C,4
130 BUHEX LD   B,4
140       LD   A,E
150       AND  15
160 BUCIF RR   D
170       RR   E
180       DJNZ BUCIF
190       CP   10
200       JR   C,NUM
210       ADD  A,7
220 NUM   ADD  A,48
230       LD   (HL),A
240       DEC  HL
250       DEC  C
260       JR   NZ,BUHEX
270       EX   DE,HL
280       LD   BC,4
290       INC  DE
300       CALL #2AB2;STACKSTO
310       RST  #28;CALCULADOR
320       DEFB #17;SUMAR STR$
330       DEFB #38;FIN
340       LD   BC,1
350       RET
360 SPARE DEFW 0,0
```

vertido en el resultado de nuestra rutina. Esta es la forma más sencilla en que podemos devolver cadenas al Basic. No podemos devolverlas directamente como resultado de USR al igual que hacíamos al devolver números porque el sistema operativo siempre espera que el resultado de USR sea un número, y no podría tratarlo como una cadena. Antes de regresar al Basic se carga un 1 en el registro BC para evitar que contenga Ø y la expresión cadena AND Ø dé otra vez la cadena vacía.

16 MULTIPLICACIÓN DE CADENAS ALFANUMÉRICAS

Longitud: 63 bytes.
Dirección: reubicable.
Forma de uso: colocar en alguna línea DEF FN M$(A$,N)=" "AND USR dirección y hacer LET X$=FN M$(A$,N).

Esta rutina no es propiamente aritmética, pero la colocamos en este grupo por su uso del calculador. Lo que hace es crear una cadena igual a N veces la cadena especificada. Por ejemplo, para llenar la pantalla de asteriscos haríamos PRINT FN M$("*",7Ø4). La rutina comienza calculando cuál va a ser la longitud total de la cadena, y almacena en la pila los datos de una supuesta cadena con esta longitud. Al sumársela a la cadena vacía, se crea en el área de trabajo una cadena de las dimensiones deseadas, que posteriormente rellenaremos con los caracteres adecuados.

LISTADO A

línea	datos	control
1	DD2A0B5CDD5E06DD5607	1001
2	D5DD6E0DDD660EE5CDA9	1497
3	30E5C1AFCDB62AEF1731	1385
4	38CDF12BDD6E04DD6605	1208
5	DDE1C1DD7CDDB5280ADD	1657
6	2BE5C5EDB0C1E118F001	1565
7	0100C90000000000000000	202

DUMP: 6Ø.ØØØ
N.° DE BYTES: 63

LISTADO B

```
10 #C-
20 #D+
30 ;
40 ; **MULTIPLICAR CADENA**
50 ;
60 ;
70 CADENA LD   IX,(23563)
80        LD   E,(IX+6)
90        LD   D,(IX+7)
100       PUSH DE
110       LD   L,(IX+13)
120       LD   H,(IX+14)
130       PUSH HL
140       CALL HLPODE
150       PUSH HL
160       POP  BC
170       XOR  A
180       CALL STKSTO
190       RST  #28;CALCULADOR
200       DEFB #17;SUMAR CADE
NAS
210       DEFB #31;DUPLICAR
220       DEFB #38;FIN
230       CALL STKFET;TOMA DI
R. DE LA CADENA
240       LD   L,(IX+4)
250       LD   H,(IX+5)
260       POP  IX
270       POP  BC
280 BUCLE DEFB 221
290       LD   A,H
300       DEFB 221
310       OR   L
320       JR   Z,NOMAS
330       DEC  IX
340       PUSH HL
350       PUSH BC
360       LDIR
370       POP  BC
380       POP  HL
390       JR   BUCLE
400 NOMAS LD   BC,1
410       RET
420 ;
430 STKSTO EQU  #2AB6
440 STKFET EQU  #2BF1
450 HLPODE EQU  #30A9
```

RUTINAS DE INTERRUPCIONES

Las interrupciones también son un apartado muy potente del Spectrum, aunque de ellas se ha hablado mucho más que del calculador. Para que una rutina de interrupciones funcione correctamente se ha de crear una tabla de ídem, que es un conjunto de 257 bytes iguales colocados en una dirección múltiplo de 256. Para activar la rutina se crea dicha tabla y se coloca en el registro I el byte alto de dicha dirección, después de lo cual se pasa al modo 2 de interrupciones. La dirección de la rutina de interrupciones deberá ser igual a 257 por el número contenido en la tabla. Todo esto significa que no podemos colocar la rutina en cualquier dirección, por lo que a diferencia de todas las anteriores, las 4 siguientes no son reubicables. En todas ellas, la tabla de interrupciones se encuentra a partir de la dirección 64768, y la rutina de interrupciones propiamente dicha en la 65278.

17 LECTURA DE TECLADO

Longitud: 97 bytes.
Dirección: 65254.
Activación: RANDOMIZE USR 65254.
Desactivación: RANDOMIZE USR 65261.

Esta rutina crea una tabla de 4Ø bytes a partir de la dirección 65311. Cada byte se corresponde con una tecla. En la figura 1 vemos de qué forma. El byte correspondiente a una tecla será 255 si la tecla no está pulsada y Ø si lo está. Esto nos servirá para hacer programas en Basic que necesiten leer la pulsación de varias teclas a la vez, o la de una tecla independiente de que

otras estén pulsadas o no, ya que como sabemos, la función INKEY$ sólo permite la lectura de una tecla cuando es la única que ha sido pulsada, y el uso de IN dificulta la lectura de una sola tecla para darnos de una vez los datos de 5.



LISTADO A

línea	datos	control
1	2100FD3EFE772C20FC24	1085
2	773DED47ED5EC93E3FED	1382
3	56ED47C9F5C5D5E50EFE	1747
4	211FFF79DBFE570605CB	1214
5	1A9F772310F9CB0138EF	1103
6	E1D1C1F1C33800008B2C	1302
7	B133000AC20B4400005C	603
8	000AC20B000AC60BE6FE	918
9	04011709C9050F3CBE3C	568
10	2B2D6533100ED000000	493

DUMP: 60.000
N.° DE BYTES: 97

LISTADO B

```
10 *C-
20 *D+
30 ;
40 ;   **LECTURA DE TECLAS**
50 ;
60 ;
70          ORG  #FEE6
80 ;
90 ACTIVA LD   HL,#FD00
100         LD   A,#FE
110 INIC    LD   (HL),A
120         INC  L
130         JR   NZ,INIC
140         INC  H
150         LD   (HL),A
160         DEC  A
170         LD   I,A
180         IM   2
190         RET
200 ;
210 DESACT LD   A,63
220         IM   1
230         LD   I,A
240         RET
250 ;
260         ORG  #FEFE
270 ;
280 LEETEC PUSH AF
290         PUSH BC
300         PUSH DE
310         PUSH HL
320         LD   C,254
330         LD   HL,TECLAS
340 BUCLE  LD   A,C
350         IN   A,(254)
360         LD   D,A
370         LD   B,5
380 TECS    RR   D
390         SBC  A,A
400         LD   (HL),A
410         INC  HL
420         DJNZ TECS
430         RLC  C
440         JR   C,BUCLE
450         POP  HL
460         POP  DE
470         POP  BC
480         POP  AF
490         JP   56
500 ;
510 TECLAS DEFS 40
```

18 FLASH DE UNA VENTANA

Longitud: 151 bytes.
Dirección: 65198.
Activación: colocar en alguna línea DEF FN F(X,Y,A,L,P,S,V)=USR 65198, y hacer RANDOMIZE FN F(X,Y,A,L,P,S,V).
Desactivación: RANDOMIZE USR 65271.

Podríamos también haber colocado esta rutina entre las de manejo de ventanas. Con ella podemos hacer un FLASH un tanto especial, ya que podemos cambiar todo lo que deseemos, y no simplemente papel por tinta y tinta por papel. X e Y son las coordenadas en baja resolución de la esquina superior izquierda de la ventana. A y L son el alto y el ancho en caracteres de la misma. P y S son los atributos que queremos que vayan alternando en el FLASH, y

LISTADO A

línea	datos	control
1	2100FD3EFE772C20FC24	1085
2	77DD2A0B5CDD7E042616	896
3	87878787CB1487CB14DD	1342
4	B60C6F223EFFDD7E2432	1089
5	40FFDD7E2C3241FFDD7E	1427
6	14321DFFDD7E1C3221FF	1067
7	DD7E343207FF3EFDED47	1334
8	ED5EC93E3FED56ED47C9	1489
9	F5C5D5E53A42FF3CFE00	1577
10	382A3A43FFEE013243FF	1089
11	2140FF2801237E2A3EFF	913
12	0600C5E501000545D1C	638
13	770D2802EDB0E10E2009	867
14	C110EBAF3242FFE1D1C1	1617
15	F1C33800000000000000	492
16	00000000000000000000	0

DUMP: 60.000
N.° DE BYTES: 151

LISTADO B

```
10 *C-
20 *D+
30 ;
40 ;   **FLASH DE VENTANAS**
50 ;
60 ;
70          ORG  #FEAE
80 ;
90 ACTIVA LD   HL,#FD00
100         LD   A,#FE
110 INIC    LD   (HL),A
120         INC  L
130         JR   NZ,INIC
140         INC  H
150         LD   (HL),A
160         LD   IX,(23563)
170         LD   A,(IX+4)
180         LD   H,22
190         ADD  A,A
200         ADD  A,A
210         ADD  A,A
220         ADD  A,A
230         RL   H
240         ADD  A,A
250         RL   H
260         OR   (IX+12)
270         LD   L,A
280         LD   (DIRIN),HL
290         LD   A,(IX+36)
300         LD   (ATTRS),A
310         LD   A,(IX+44)
320         LD   (ATTRS+1),A
330         LD   A,(IX+20)
340         LD   (FILAS+1),A
350         LD   A,(IX+28)
360         LD   (COLUS+1),A
370         LD   A,(IX+52)
380         LD   (VELP+1),A
390         LD   A,#FD
400         LD   I,A
410         IM   2
420         RET
430 ;
440 DESACT LD   A,63
450         IM   1
460         LD   I,A
470         RET
480 ;
490 ;
500         ORG  #FEFE
510 ;
520 FLASH   PUSH AF
530         PUSH BC
540         PUSH DE
550         PUSH HL
560         LD   A,(CONT)
570         INC  A
580 VELP    CP   0
590         JR   C,ESPERA
600         LD   A,(FLAGSH)
610         XOR  1
620         LD   (FLAGSH),A
630         LD   HL,ATTRS
640         JR   Z,YALIS
650         INC  HL
660 YALIS   LD   A,(HL)
670         LD   HL,(DIRIN)
680 FILAS   LD   B,0
690 BUFIL   PUSH BC
700         PUSH HL
710 COLUS   LD   BC,0
720         LD   D,H
730         LD   E,L
740         INC  E
750         LD   (HL),A
760         DEC  C
770         JR   Z,UNCOL
780         LDIR
790 UNCOL   POP  HL
800         LD   C,32
810         ADD  HL,BC
820         POP  BC
830         DJNZ BUFIL
840         XOR  A
850 ESPERA LD   (CONT),A
860         POP  HL
870         POP  DE
880         POP  BC
890         POP  AF
900         JP   56
910 ;
920 DIRIN  DEFW 0
930 ATTRS  DEFW 0
940 CONT   DEFB 0
950 FLAGSH DEFB 0
```

se construyen de la forma ya conocida. V es la velocidad del cambio de un atributo a otro, siendo 1 la más rápida y 255 la más lenta. De esta forma podemos conseguir un tipo de FLASH semejante al de los ordenadores Amstrad CPC, en los cuales podemos hacer flashes de tinta o de papel por separado.

19 RELOJ

Longitud: 195 bytes.
Dirección: 65172.
Activación: colocar en alguna línea DEF FN R(H,M)=USR 65172, y hacer RANDOMIZE FN R(H,M).
Desactivación: RANDOMIZE USR 65213.

H es la hora entre Ø y 23 y M los minutos entre Ø y 59. Una vez activada aparecerá un reloj en la esquina superior derecha de la pantalla, que sólo se detendrá cuando se ejecute algún BEEP o alguna instrucción del cassette, pues éstas inhabilitan las interrupciones.

La rutina usa su propia subrutina de impresión, ya que usar la de la ROM durante las interrupciones causaría problemas. Para poder calcular el tiempo con exactitud, basta con que sepamos que las interrupciones se producen 5Ø veces cada segundo, por lo que tendremos que ir contando cuantas interrupciones se van produciendo, y cada vez que lleguemos a 5Ø, contar un segundo más. Luego, cuando lleguemos a 6Ø segundos contaremos un minuto, etc. La rutina de impresión utiliza la variable del sistema CHARS, por lo que si la modificamos podremos hacer que las cifras del reloj se impriman con caracteres definidos por nosotros.

LISTADO A

línea	datos	control
1	2100FD3EFE772C20FC24	1085
2	77DD2A0B5CDD7E0432FA	1136
3	FEDD7E0C32FBFEAF32FC	1645
4	FE32FDFE3EFDED47ED5E	1765
5	C93E3FED56ED47C90630	1212
6	D60A380304 18F94F78CD	964
7	DBFE79C63ACDDBFEC93E	1791
8	3A87876F260029ED5B36	900
9	5C19ED5BF8FE7B3C32F8	1428
10	FE06087E23121410FAC9	934
11	000000000000F5C5D5E5	884
12	3AFDFE3CFE3220253AFC	1308
13	FE3CFE3C20193AFBFE3C	1308
14	FE3C200D3AFAFE3CFE18	1259
15	2001AF32FAFEAF32FBFE	1492
16	AF32FCFEAF32FDFE2118	1520
17	4022F8FE3AFAFECDC4FE	1817
18	CDD9FE3AFBFECDC4FECD	2099
19	D9FE3AFCFECDC4FEE1D1	2124
20	C1F1C338000000000000	685

DUMP: 60.000
N.° DE BYTES: 195

LISTADO B

```
10 *C-
20 *D+
30 ;
40 ;        **RELOJ**
50 ;
60 ;
70          ORG  #FE94
80 ;
90 ACTIVA LD   HL,#FD00
100         LD   A,#FE
110 INIC    LD   (HL),A
120         INC  L
130         JR   NZ,INIC
140         INC  H
150         LD   (HL),A
160         LD   IX,(23563)
170         LD   A,(IX+4)
180         LD   (HORA),A
190         LD   A,(IX+12)
200         LD   (MIN),A
210         XOR  A
220         LD   (SEG),A
230         LD   (CINC),A
240         LD   A,#FD
250         LD   I,A
260         IM   2
270         RET
280 ;
290 DESACT  LD   A,63
300         IM   1
310         LD   I,A
320         RET
330 ;
340 NUM     LD   B,48
350 BUC     SUB  10
360         JR   C,ENCA
370         INC  B
380         JR   BUC
390 ENCA    LD   C,A
400         LD   A,B
410         CALL PRIN
420         LD   A,C
430         ADD  A,58
440         CALL PRIN
450         RET
460 ;
470 PRINDP  LD   A,":"
480 PRIN    ADD  A,A
490         ADD  A,A
500         LD   L,A
510         LD   H,0
520         ADD  HL,HL
530         LD   DE,(23606)
540         ADD  HL,DE
550         LD   DE,(STPRT)
560         LD   A,E
570         INC  A
580         LD   (STPRT),A
590         LD   B,8
600 BPR     LD   A,(HL)
610         INC  HL
620         LD   (DE),A
630         INC  D
640         DJNZ BPR
650         RET
660 ;
670 STPRT   DEFW 0
680 HORA    DEFB 0
690 MIN     DEFB 0
700 SEG     DEFB 0
710 CINC    DEFB 0
720 ;
730         ORG  #FEFE
740 ;
750 RELOJ   PUSH AF
760         PUSH BC
770         PUSH DE
780         PUSH HL
790         LD   A,(CINC)
800         INC  A
810         CP   50
820         JR   NZ,NOMA1
830         LD   A,(SEG)
840         INC  A
850         CP   60
860         JR   NZ,NOMA2
870         LD   A,(MIN)
880         INC  A
890         CP   60
900         JR   NZ,NOMA3
910         LD   A,(HORA)
920         INC  A
930         CP   24
940         JR   NZ,NOMA4
950         XOR  A
960 NOMA4   LD   (HORA),A
970         XOR  A
980 NOMA3   LD   (MIN),A
990         XOR  A
1000 NOMA2  LD   (SEG),A
1010        XOR  A
1020 NOMA1  LD   (CINC),A
1030        LD   HL,16408
1040        LD   (STPRT),
HL
1050        LD   A,(HORA)
1060        CALL NUM
1070        CALL PRINDP
1080        LD   A,(MIN)
1090        CALL NUM
1100        CALL PRINDP
1110        LD   A,(SEG)
1120        CALL NUM
1130        POP  HL
1140        POP  DE
1150        POP  BC
1160        POP  AF
1170        JP   56
```

20 SCROLL DE UN MENSAJE POR LA PANTALLA

Longitud: 168 bytes.
Dirección: 652ØØ.
Activación: colocar en alguna línea DEF FN S(D,L)=USR 652ØØ, y hacer RANDOMIZE FN S(D,L).
Desactivación: RANDOMIZE USR 6526Ø.

D es una dirección a partir de la que previamente habremos colocado los códigos ASCII del mensaje deseado, finalizando con un 255. L es la línea en la que queremos que aparezca el mensaje que puede ser cualquiera entre Ø y 23.

La rutina utiliza las siguientes variables internas:

— DIRSC. Dirección de memoria del extremo derecho de la línea en la que queremos efectuar el scroll. Es calculada al activar la rutina y almacenada en DIRSC para su posterior uso.

— CONSC. Contador de scrolls. Sirve para saber cuándo hemos efectuado 8 scrolls de un pixel y debemos, por tanto imprimir otra letra.

— TEXIN. Dirección de memoria donde se encuentra el comienzo del texto a imprimir.

— TEXAC. Dirección del carácter del texto que debemos imprimir a continuación se inicializa originalmente en la activación con el mismo valor que TEXIN.

— BUFF. No es exactamente una variable, sino un espacio de 8 bytes en el que almacenamos el carácter que va a aparecer en la pantalla.

El procedimiento de la rutina es el siguiente:

— Incrementar (CONSC).

— Si no ha llegado a ocho, pasar directamente al proceso de scroll.

— Se toma el contenido de la dirección señalada por TEXAC.

— Si es 255, se vuelve a inicializar TEXAC con TEXIN para que el texto se repita desde el principio.

— Tanto si se ha inicializado como si no, se incrementa TEXAC.

— Se calcula la dirección de memoria del carácter correspondiente y se trasladan sus 8 bytes a BUFF.

— Aquí comienza el proceso del scroll. Cargamos HL con (DIRSC) y DE con BUFF, o lo que es lo mismo, HL señala a la primera línea de pixels de la fila correspondiente de la pantalla y DE a la primera línea de pixels del carácter que está entrando en la pantalla.

— Ahora entramos en un bucle que se repetirá ocho veces. Cada pasada por el bucle efectuará el scroll de una de las ocho líneas de pixels que tiene de alto un carácter. Dentro del bucle:

— Giramos o «scrollamos» el carácter almacenado en BUFF. De esta manera queda en el bit de acarreo el pixel que ha de entrar en la pantalla.

— Realizamos 31 veces un bucle en el que vamos desplazando toda la línea. El uso de RL pasa al bit de acarreo el pixel que ha de pasar al byte siguiente (en realidad el anterior, ya que estamos haciendo un scroll hacia la izquierda), a la vez que toma del acarreo el pixel que viene del byte anterior (en realidad el siguiente).

— Desplazamos el último byte por la izquierda.

— Alzamos los cinco bytes inferiores de L para que HL vuelva a señalar al extremo derecho.

— Hacemos que tanto HL como DE señalen a la siguiente línea de pixels para que estén listos para una nueva pasada por el bucle.

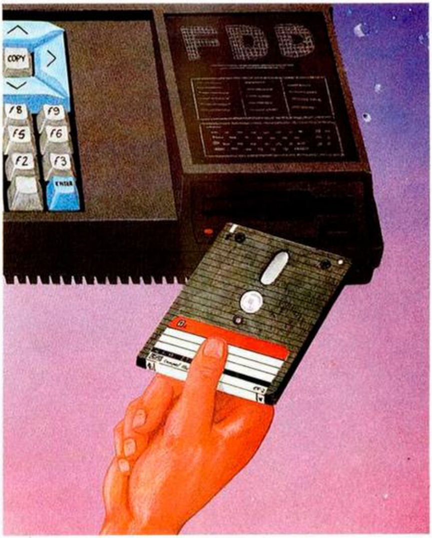
LISTADO A

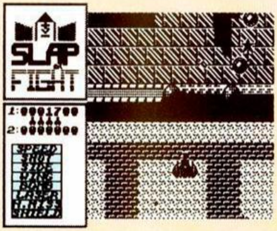
línea	datos	control
1	2100FD3EFE772C20FC24	1085
2	773E0732F3FEDD2A0B5C	1101
3	DD6E04DD66052254FF22	1070
4	56FFDD7E0C47E6070F0F	1038
5	0FF61F6F78E618F64067	1190
6	22F4FE3EFDED47ED5EC9	1687
7	3E3FED56ED47C9000000	957
8	FAC900000000000000F5C5	893
9	D5E53AF3FE3CE60732F3	1587
10	FE20232A54FF7EFEFF20	1369
11	042A56FF7E232254FF26	959
12	006F292929ED5B365C19	733
13	11F6FE010800EDB02AF4	1225
14	FE11F6FE0E08EBCB16EB	1488
15	061FCB162D10FBCB167D	924
16	F61F6F24130D20EAE1D1	1156
17	C1F1FFC90000000000000	890

DUMP: 60.000
N.° DE BYTES: 168

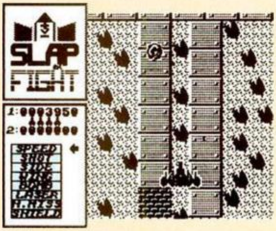
LISTADO B

```
10 *C-
20 *D+
30 ;
40 ;   **SCROLL DE TEXTO**
50 ;
60 ;
70         ORG  #FEB0
80 ;
90 ACTIVA LD   HL,#FD00
100        LD   A,#FE
110 INIC   LD   (HL),A
120        INC  L
130        JR   NZ,INIC
140        INC  H
150        LD   (HL),A
160        LD   A,7
170        LD   (CONSC),A
180        LD   IX,(23563)
190        LD   L,(IX+4)
200        LD   H,(IX+5)
210        LD   (TEXAC),HL
220        LD   (TEXIN),HL
230        LD   A,(IX+12)
240        LD   B,A
250        AND  7
260        RRCA
270        RRCA
280        RRCA
290        OR   31
300        LD   L,A
310        LD   A,B
320        AND  24
330        OR   64
340        LD   H,A
350        LD   (DIRSC),HL
360        LD   A,#FD
370        LD   I,A
380        IM   2
390        RET
400 ;
410 DESACT LD   A,63
420        IM   1
430        LD   I,A
440        RET
450 ;
460 CONSC  DEFB 0
470 DIRSC  DEFW 0
480 BUFF   DEFS 8
490 ;
500        ORG  #FEFE
510 ;
520 SCRTEX PUSH AF
530        PUSH BC
540        PUSH DE
550        PUSH HL
560        LD   A,(CONSC)
570        INC  A
580        AND  7
590        LD   (CONSC),A
600        JR   NZ,NONULE
610        LD   HL,(TEXAC)
620        LD   A,(HL)
630        CP   255
640        JR   NZ,NOPROB
650        LD   HL,(TEXIN)
660        LD   A,(HL)
670 NOPROB INC  HL
680        LD   (TEXAC),HL
690        LD   H,0
700        LD   L,A
710        ADD  HL,HL
720        ADD  HL,HL
730        ADD  HL,HL
740        LD   DE,(23606)
750        ADD  HL,DE
760        LD   DE,BUFF
770        LD   BC,8
780        LDIR
790 NONULE LD   HL,(DIRSC)
800        LD   DE,BUFF
810        LD   C,8
820 BUSCR  EX   DE,HL
830        RL   (HL)
840        EX   DE,HL
850        LD   B,31
860 ROSCAN RL   (HL)
870        DEC  L
880        DJNZ ROSCAN
890        RL   (HL)
900        LD   A,L
910        OR   31
920        LD   L,A
930        INC  H
940        INC  DE
950        DEC  C
960        JR   NZ,BUSCR
970        POP  HL
980        POP  DE
990        POP  BC
1000       POP  AF
1010       RST  56
1020       RET
1030 ;
1040 TEXAC DEFW 0
1050 TEXIN DEFW 0
```

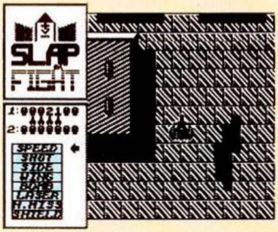




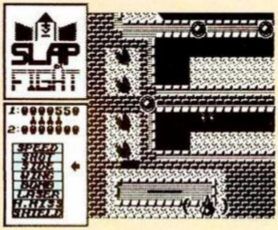
Al comenzar el juego lo primero que llama nuestra atención es un curioso conjunto de nombres en la esquina inferior de la pantalla. Cada palabra tiene un significado. No es necesario pulsar continuamente la tecla de disparo para utilizar las armas, ya que si dejamos el dedo puesto en el disparo el fuego es constante.



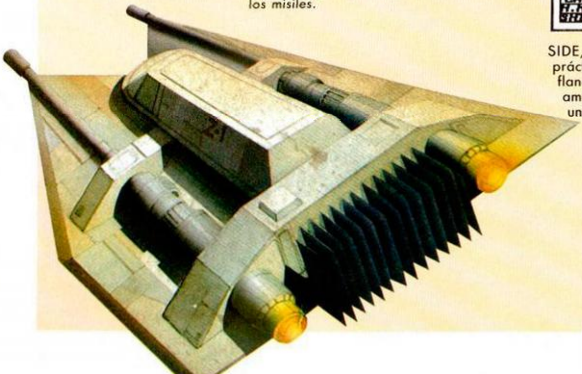
Al eliminar alguno de los enemigos aparecen unas estrellas como la que se observa en la foto anterior. Al recogerla en el menú de opciones aparece una flecha junto a la palabra SPEED, con ella conseguimos que el movimiento de la nave sea dos veces superior, para ello pulsaremos la tecla space.



Si no utilizamos ésta tecla, al recoger otra estrella la flecha pasará a la siguiente palabra y así hasta que utilicemos alguna de ellas. SHOT, nos permite utilizar los misiles.



SIDE, hace que nuestra nave sea prácticamente inalcanzable por los flancos. Cuando disparamos, a ambos lados de la nave aparece una barrera protectora.



El intrépido piloto a los mandos de su sofisticada nave se dispuso a introducirse en el mundo oculto del megabús. Este se encontraba repleto de pequeños bytes degenerados por la terrible explosión de la Rom, que se produjo al recalentarse la central de la Ula.

Este juego, como la mayoría de los que últimamente aparecen en el mercado (Xevious, 1942, etc.) es similar a los de las máquinas de juego de los bares. Para ayudaros a llegar al final, hemos preparado unos cargadores y unos pequeños consejos que os permitirán acabar el juego *con una gran puntuación.*

SPECTRUM

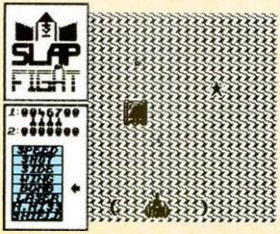
SLAP FIGHT



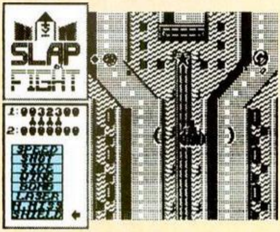
Nuestra pequeña nave puede llegar a convertirse en un arsenal volante utilizando la opción del menú denominada WING. Cada vez que pulsemos la tecla space dentro de esta opción, nuestra nave incorporará una pareja de lanzaderas más a sus flancos.



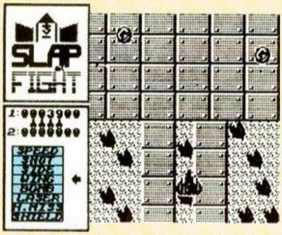
Podemos triplicar el tamaño de la nave, colocando el cursor en la opción WING. También hemos de tener en cuenta que al ser mayor nuestra nave más fácilmente puede ser alcanzada por un disparo enemigo.



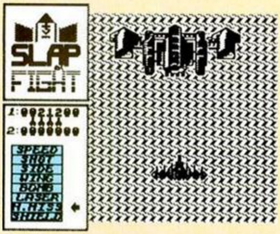
BOMB, como su propio nombre indica, bomba, hace que nuestra lanzadera despida unas poderosas bombas capaces de destruir todo aquello que se encuentre al alcance de su terrible onda expansiva.



En casos de apuro un arma secreta hará que recorramos una gran distancia del mapa. En el menú se encuentra en último lugar y su nombre es SHIELD. En la foto no podemos ver su estupendo poder, pero al utilizarla observaréis que el borde de la pantalla empieza a parpadear y hasta que esto termine nuestra nave es indestructible sea cual sea el tipo de enemigo al que nos enfrentemos. Este arma se desgasta por el choque con enemigos o con cualquiera de sus disparos, así como por la cantidad de éstos que nosotros realicemos.



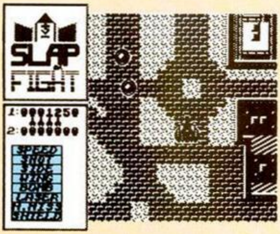
Los paisajes se van generando a nuestro paso, siendo de lo más pintorescos a medida que nos adentramos en el juego.



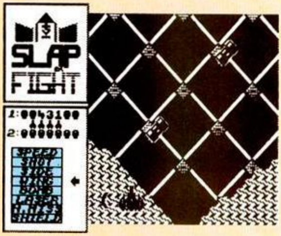
Todos los enemigos son eliminados con cualquiera de las armas, ya sean bombas, misiles o láseres, pero cada uno de los enemigos es más vulnerable a una de ellas.

FASES DE LA NAVE





La mayoría de nuestros enemigos están representados por unos platillos volantes que se desplazan con movimientos siempre iguales por la pantalla, disparando al punto en donde se encuentre nuestra nave, por lo que es aconsejable no permanecer quieto demasiado tiempo.



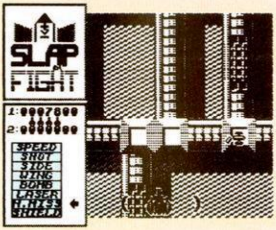
Uno de los enemigos más incómodos son una especie de cilindros móviles que se desplazan entre las vigas del suelo. Al destruirlos aparece un nuevo enemigo.

SPECTRUM

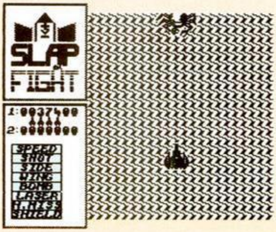
```
  10 BORDER 0: PAPER 0: CLEAR 25
000
  20 LOAD "scr"SCREEN$ : LOAD ""
CODE
  30 POKE 57175,201: RANDOMIZE U
SR 48400
```

ENEMIGOS

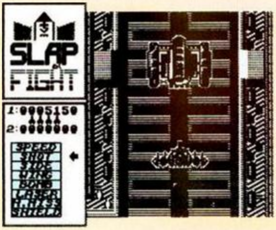




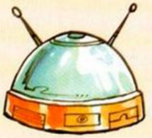
Al llegar a este punto del juego es aconsejable el uso de SHIELD, ya que unas pequeñas naves con forma de mosquitos se colocarán en línea recta burlando nuestros ataques, y al posarse en una de las plataformas dispararán en todas las direcciones. La única forma de destruirlas es en ese momento o, con un poco de suerte, dispararlas justo cuando el centro de la nave esté en el blanco.



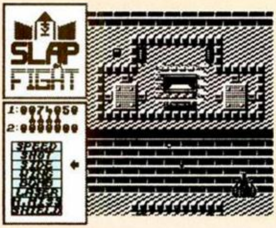
Los alacranes mecánicos los destruiría la abuelita poke con la mirada, pero no es imprescindible. Hasta nosotros, torpes humanos, hemos encontrado un sistema más fácil para acabar con tan asquerosa especie de reptiles. Basta con que coloquéis vuestra nave en la parte izquierda de la pantalla y a media altura, en cuanto veáis aparecer el primero poneros un poco más a la izquierda de donde apareció éste y disparar continuamente, no aparecerá otro, ya que se destruirá antes de salir en la pantalla.



Al término de la primera fase, en una especie de autopista en la que no aparece ningún enemigo, una inmensa nave subirá y bajará recorriendo la pantalla disparando continuamente.







Al final del juego, para sobrepasar las defensas, haremos movimientos circulares alrededor de éstas, disparando misiles teledirigidos y teniendo en cuenta que cuando empiezan a desaparecer éstas no disparan más. A continuación una nueva defensa aparece por la parte superior de la pantalla por lo que deberemos bajar.



Pablo Ariza

CREADOR DE JUEGOS ARCADE



Lo que presentamos hoy es una modificación y adaptación de la anterior rutina de sprites. Los que no tengan el MICROHOBBY Especial número 6 no deben preocuparse, pues incluimos también en este número dicha rutina. Pero los que sí lo tengan y hayan tecleado los listados publicados en él no necesitarán teclear el listado 2 o el 5, pues equivalen respectivamente a los listados 3 y 4 del artículo anterior, así que cada vez que se haga una referencia a dichos listados utilizarán los que ya tienen grabados.

Evidentemente, antes de poder usar el programa, necesitamos tener unos gráficos que dibujar. Para diseñar vuestros gráficos podéis usar uno de los muchos programas de dibujo disponibles comercialmente. Para los gráficos de la demostración que acompaña a este artículo se ha usado el Edigraf, publicando en el MICROHOBBY Especial número 4.

Como ya explicamos en el artículo del número pasado, los sprites constan de dos gráficos: el gráfico propiamente dicho y su máscara. Ambos deben tener las mismas dimensiones y estar almacenados consecutivamente en la memoria. En la máscara deberemos llenar con tinta las zonas del gráfico que sean transparentes. Si un sprite consta de varias fases de animación almacenaremos las fases una detrás de otra, y cada gráfico con su máscara detrás.

Podríamos dar muchos más detalles sobre la forma en que se crean y almacenan los gráficos y máscaras, pero ya hay mucha literatura sobre el tema y será mejor que en este artículo no entremos en más explicaciones y nos dediquemos al tema que nos ocupa. Tan sólo diremos que los gráficos a usar por las rutinas expuestas aquí deben ser almacenados scan tras scan, es decir, primero todos los bytes del primer scan de izquierda a derecha, luego los del segundo, etc.

Características de la rutina

Con la rutina presentada en este artículo podemos definir un mapa de un máximo de 255 por 255 caracteres, con un número de objetos sólo limitado por la cantidad de memoria y la velocidad. Todos los objetos existentes en el mapa serán de una de las 255 clases de decorados que podemos definir como máximo. Cada una de estas clases de decorados tendrá un gráfico y un color determinado (sólo un color por decorado) y unas dimensiones cualesquiera con la única condición de que sean menores que las de la pantalla del ordenador (24 por 32 caracteres).

En el anterior número de MICROHOBBY Especial publicamos una rutina que permitía controlar sprites desde nuestros propios programas en Basic o C/M. Ahora no sólo podréis dibujar los sprites, sino también los decorados y los fondos de vuestros juegos, y moverlos con un suave scroll, todo ello con gran facilidad gracias a este programa monitor.

Podemos definir el tamaño de la ventana en la que queremos que aparezcan los gráficos en la pantalla, lo que nos permite reservar una zona de la pantalla del ordenador para marcadores o lo que queramos. Esto no era posible con la rutina de sprites anterior.

Tenemos un máximo de 17 sprites cada uno con las dimensiones, fases de animación, color, velocidad, etc., que queramos. Además del movimiento en línea recta podemos definir trayectorias preprogramadas para que las sigan los sprites que queramos, y también podemos controlar directamente el movimiento con nuestro programa.

Existe la posibilidad de detectar el choque entre dos sprites o entre un sprite y cualquiera de los restantes.

"Con la rutina presentada en este artículo podemos definir un mapa de un máximo de 255 por 255 caracteres"



Queda a cargo de nuestro programa la acción que queramos llevar a cabo cuando el choque se produzca. Igualmente se puede detectar el choque entre un sprite y un objeto del mapa, bien sea uno cualquiera o de una clase determinada.

Programa monitor

Para facilitar al máximo la introducción de todos los datos necesarios para definir el mapa de un juego, hemos desarrollado un programa Basic que se encarga de pedirnos y procesar toda la información requerida. Para poder usarlo tendremos que teclear en primer lugar el listado 1 y grabarlo con LINE 1. A continuación tecleamos el listado 2 en el cargador hexadecimal o bien el listado 5 en un ensamblador (o los listados equivalentes del número pasado) y grabarlo como "CODSPRIT". Por último teclearemos el listado 3 o el 4 y lo grabaremos como "CODMAP". Para usar el programa lo cargaremos todo desde el principio.

El programa nos permite cargar un bloque de bytes en el que se encuentren todos los gráficos que vayamos a utilizar (ya sean de sprites o de decorados). Nos preguntará qué longitud ocupa el bloque y en que dirección fueron creados. Cuando hayamos terminado la introducción de datos, el programa trata de colocar los gráficos lo más alto posible en la memoria, justo delante de las rutinas y datos de mapas y sprites. De esta forma conseguimos aprovechar al máximo la memoria. Al mover el bloque de los gráficos cambiarán las direcciones en que se encuentran éstos, así que el programa se encarga automáticamente de calcular la nueva dirección de ca-

da gráfico a partir de su dirección antigua, la dirección original donde comenzaban los gráficos y la nueva dirección de comienzo.

Tanto si hemos cargado gráficos como si no, pasaremos después a lo que hemos llamado definición de formatos, por llamarlo de alguna forma. Son una serie de determinadas posibilidades de las rutinas. Los datos que pide, comentados, son los siguientes:

Datos del mapa.

— Color de fondo (+8*brillo). El color se refiere al papel, puesto que se trata de un color de fondo. Al especificar más adelante el color de los sprites, sólo se dará la tinta, y se utilizará el papel especificado aquí.

— Ancho en caracteres. Ancho total del mapa. Sirve para delimitar los bordes del mismo. Cuando el sprite principal llega a un borde no puede continuar avanzando.

— Alto en caracteres. Ídem que para el ancho.

— Velocidad del scroll. Es la cantidad de pixels que se moverán en el decorado cada vez que avancemos. Preferiblemente un múltiplo de 2; los mejores valores están entre 2 y 16. Cuanto mayor sea el valor, mayor la velocidad, pero menor la suavidad. Es importante saber qué velocidad de scroll vamos a usar al ponernos a diseñar los gráficos, pues deberemos dejar alrededor de cada gráfico un margen de al menos «n» pixels en blanco, siendo n la velocidad del scroll. Esto sirve para evitar tener que borrar los gráficos antiguos antes de dibujar los nuevos. Si no dejamos dicha franja en blanco, al moverse el mapa los gráficos irán dejando rastros.

Datos de la pantalla. En esta sección se nos piden las dimensiones de la ventana que queremos que se vuelque en pantalla. Puede ir desde toda la pantalla hasta un solo carácter. Los datos necesarios son la fila y la columna de la esquina superior izquierda y el ancho y el alto, todo ello en caracteres. Nótese que la ventana puede contener parte o la totalidad de las dos líneas inferiores de la pantalla.

Datos del sprite principal. El sprite principal es aquél sobre el que se va a producir el scroll, es decir dicho sprite quedará fijo en la pantalla mientras que los fondos del mapa y el resto de los sprites se mueven a su alrededor.

— Coordenadas X, Y. Coordenadas de la pantalla del ordenador en las que queremos aparezca dibujado el sprite. Si las coordenadas que especificamos están fuera de la ventana definida previamente, dicho sprite no aparecerá.

— Código. Código o valor de prioridad (Ø-16) del sprite que vamos a considerar como principal. Será el código del sprite del personaje central del juego.

— Dir de los gráficos derecha, abajo, etc. Son las direcciones donde tenemos almacenados los gráficos del sprite principal avanzando en dichas direcciones. Cuando posteriormente especificamos la dirección en la que queremos el scroll del mapa, el sprite se dibujará con los gráficos correspondientes. Si sólo vamos a hacer, por ejemplo un scroll de cuatro direcciones, no necesitaremos las direcciones de los gráficos en diagonal, así que cuando se nos pida la primera de éstas introduciremos un Ø y ya no se nos pedirán más.

En cada una de las secciones se nos da la posibilidad de repetir los datos si nos hemos equivocado. Una vez introducidos todos estos datos previos pasamos al menú principal. En este hay varias opciones que veremos una a una:

Repetir formatos. Nos da otra oportunidad para introducir todos los datos anteriores por si hemos cometido un error o, simplemente, hemos cambiado de idea.

Definir sprites. Por cada sprite se nos pedirán los siguientes datos:

— Ancho en caracteres. Si se introduce Ø, se volverá al menú principal.

— Alto en pixels.

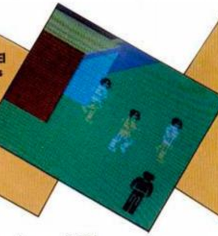
— Dirección del gráfico. Dirección donde se encuentra el primer o único gráfico de los que componen el sprite.

— Fases de animación. 1 si no tiene.

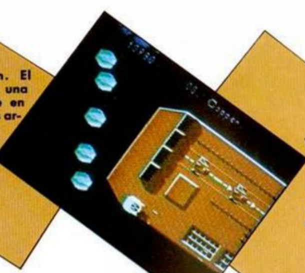
— Si en más de dos fases de animación se nos pide elegir, entre cíclica o adelante/atrás. En la primera al terminar la última fase, comienza de nuevo la primera. En la segunda, al llegar a la última fase, se repiten todas de nuevo hacia atrás, luego otra vez hacia adelante, etc.

— Color (+8*brillo). Sólo de tinta. Los sprites no tienen color propio de papel.

— Coordenadas iniciales. Pueden valer entre -32768 y +32767. Nótese que ahora no son coordenadas referidas a la panta-



Match Day. El 100% de los juegos deportivos utilizan scroll.



Uridium. El scroll es una constante en los juegos arcade.



Timanog. No es necesario ocupar toda la pantalla para conseguir un buen movimiento.

lla, como sucedía con la rutina de sprites por sí sola, sino referidas al mapa. Si la coordenada X es mayor de 32511, el sprite se considerará desactivado.

— Movimiento en línea recta. Nos permite elegir entre movimiento en línea recta o siguiendo una trayectoria preprogramada.

Si en línea recta:

— Incrementos X e Y (-128 a +127).

Si trayectoria preprogramada:

— Quieres definir la trayectoria. Nos permite definir la trayectoria que queramos.

Si contestamos no:

— Dir de la trayectoria. Dirección donde comienzan los datos de la misma.

Si contestamos sí:

— Incremento X (o dir gráfico). Incremento X (-128 a +127) a efectuar a la coordenada X, o dirección del gráfico con que queremos que se dibuje el sprite.

— Incremento Y. Este proceso se repite hasta que demos 127 como incremento X. Después se nos informará de dónde ha quedado colocada la trayectoria, lo que nos servirá para su posterior uso.

Definir decorados. Esta opción nos permite definir las clases distintas de decorados que podrá haber en el mapa. Cada decorado tendrá asociado un código, que es el que especificaremos al definir en el mapa un objeto de esa clase. Para cada decorado especificamos la dirección del gráfico, las dimensiones y el color.

Definir mapa. Aquí es donde realmente definimos los objetos existentes en el mapa. Para ello damos un código de decorado y unas coordenadas en caracteres.

Grabarlo todo. Al elegir esta opción se colocan los gráficos (si se han cargado) en la posición más alta posible, según los decorados, objetos y trayectoria que hayan sido definidos. También se hacen las actualizaciones de todas las direcciones de gráficos especificadas anteriormente. A continuación nos pide un nombre (máximo nueve caracteres) y nos graba las rutinas junto con todos los datos y los gráficos en dos bloques. Hemos de apuntar sus comienzos y longitudes, que aparecerán en la pantalla, para saber por debajo de qué dirección de memoria queda libre para nuestros programas. Toda la memoria comprendida entre la dirección del primer bloque y el final de la memoria está ocupada por los gráficos, las rutinas, datos, tablas, variables y la pantalla de trabajo, así que nuestros programas deberán estar por debajo. Hay que advertir que tras elegir la opción de grabarlo todo, podemos volverlo a grabar cuantas veces queramos, pero no podemos continuar con la definición, así que sólo debemos usarla al haber finalizado nuestro trabajo. En principio, no es posible grabar nuestro trabajo a medias, pero podemos lograrlo interrumpiendo el programa cuando estemos en el menú y salvándolo con cuidado de no borrar las variables. A continuación grabaremos toda la memoria por encima de la dirección 40000, es decir, CODE 40000,25536. Para reanudar el trabajo otro día haremos:

CLEAR 39999:LOAD"":LOAD""CODE:GO TO 170

Nuevo mapa. La rutina está preparada para tener almacenados hasta ocho mapas y usar cada vez el que se quiera. Cuando hayamos terminado de definir un mapa y queramos comenzar otro, elegiremos esta opción.

Utilización desde nuestros programas

Una vez que hemos creado todos los datos con el programa monitor o de cualquier otra forma; (el programa monitor es una ayuda, no es indispensable para el uso de la rutina), podemos cargar los dos bloques de bytes creados y usarlos en nuestros propios programas. Una vez cargados:

- Para activarlo todo debemos hacer:

POKE 52966,N:RANDOMIZE USR 51900:POKE 23681,M

N es el número de mapa





Night Shade. La Complejidad de un mapa con scroll alcanza un grado maximo en este programa.



Spy Hunter. Fue uno de los pioneros en la utilización del scroll, luego otros le imitaron.

(Ø-7), y M la cantidad de sprites definidos. A partir de ahora podemos controlar la dirección en que se mueve el sprite principal pokeando la dirección 52969 con un número entre Ø y 8. La figura 1 muestra a qué movimiento corresponde cada uno.

• Para detectar el choque entre dos sprites hacemos:

POKE 23729,A:POKE 23728, B:IF USR 52684 THEN REM HAN CHOCADO

• Para detectar el choque entre un sprite y otro cualquiera hacemos b=255.

• Para detectar el choque entre un sprite y un objeto del mapa hacemos:

POKE 23729,O:
POKE 23728,S: LET C=USR 5296Ø

S es el código del sprite. Si O es 255 se comprobará el choque con cualquier objeto del mapa, en otro caso, sólo se comprobarán los objetos cuya clase o código de decorado sea O. C terminará valiendo 65535 si no hay choque. En otro caso, será el número de elemento dentro de la tabla de objetos del mapa con el que se ha producido el choque.

• Para desactivar la ruti-

Terra Cresta. La información del juego se observa con multiscroll.



Hysteria. Es uno de los juegos que incorpora la técnica multiscroll.



Freddy Hardest. Los juegos de Dinamic suelen tener alguna fase con esta técnica



FIG. 2
MAPA DE MEMORIA

Dirección	Nombre	Descripción
?		Gráficos (de decorados y de sprites)
(TAMAPA)		Datos del mapa
(TADECO)		Datos de los decorados
?		Trayectorias prefijadas de los sprites
519ØØ	INIMAP	Entrada principal de activación de la rutina
52132	RELASU	Transforma coordenadas absolutas en relativas
5219Ø	MAPA	Subrutina principal de dibujo del mapa
52552	DIBMAP	Dibuja un elemento del mapa
52668	TAUME	Tabla de aumentos (usada para la dirección del scroll)
52684	NCOMCH	Nueva entrada a COMCHOC
5269Ø	CHODEC	Comprobación choque sprite-objeto
52927	TAMAPA	Puntero a mapa actual
52929	NUMDEF	Objetos existentes en el mapa actual
52931	TADECO	Puntero a tabla de decorados
52933	NUMDEC	Decorados distintos definidos
52934	TATABS	Tabla de direcciones de los mapas
5295Ø	TADEFS	Tabla de número de elementos de los mapas
52966	MAPACT	Mapa actual
52967	BACK	Color del fondo
52968	MAIN	Sprite principal
52969	SCROLL	Dir del scroll
5297Ø	MAPX	Ancho del mapa (caracteres)
52971	MAPY	Alto del mapa (caracteres)
52972	COLUMN	Columna izquierda de la ventana
52973	FILA	Fila superior de la ventana
52974	WIDSC	Ancho de la ventana (caracteres)
52975	HIGHSC	Alto de la ventana (caracteres)

Dirección	Nombre	Descripción
52976	CALCUL	Valor usado por la subrutina de volcado
52977	FILAPE	Coordenada Y en la pantalla del sprite principal
52978	COLUPE	Coordenada X en la pantalla del sprite principal
52979	DIRPEØ	Dirección de los gráficos hacia la derecha
.	.	.
.	.	.
52993	DIRPE7	Dirección de los gráficos hacia arriba a la derecha
52995	VESCRO	Velocidad en pixels del scroll
52996	XPER	Variables internas. Coordenadas relativas a la esquina superior izquierda de la pantalla
52998	YPER	
53ØØØ	TASPRI	Tabla de datos de los sprites
53289	TABLDI	Tabla de los sprites que se van a dibujar en pantalla
53544	SPARES	Espacio para guardar los trozos de pantalla
54568(7)	ORIGSC	Pantalla de memoria
6Ø9Ø4	ORIGAT	Atributos de memoria
61696		Tabla de interrupciones
61953	INICIO	Activación de sprites (llamada por INIMAP)
62Ø75	DESACT	Desactivación de la rutina
62Ø82	COMCHO	Comprobación de choque
62194	ENTINT	Entrada principal de interrupciones
63488		Tabla de rotaciones

na haremos RANDOMIZE USR 62075.

Con esta información y mucha imaginación se pueden hacer grandes cosas sin demasiados esfuerzos programativos.

Demostración

Para que podáis ver una pequeña muestra de lo que se puede hacer con estas rutinas sin necesidad de empezar a diseñaros vuestros gráficos y vuestro mapa, adjuntamos unos listados de demostración. Para ver la demo, hay que teclear primero el listado 6 y grabarlo con LINE 9999. A continuación grabaremos los ya tecleados listados 2 (ó 5) y 3 (ó 4). Por último, teclearemos y grabaremos los listados 7 y 8 en ese orden. Los datos del mapa y del sprite de esta demostración han sido creados con el programa monitor. Los gráficos se han hecho con el programa Edigraf antes mencionado. Controlamos una especie de tanque con las teclas de los cursores (5, 6, 7 y 8). Tan sólo como una burda muestra de la utilidad que pueden tener las rutinas de detección de choques veremos cómo si nos abalanzamos contra un árbol yendo de abajo a arriba, éste será sustituido por un árbol «roto». Aunque el efecto no está muy bien logrado, ilustra muy bien qué tipo de cosas se pueden hacer fácilmente. Seguro que sin mucho esfuerzo, cualquiera de vosotros podría hacerlo mucho mejor. Para acabar la demostración basta con pulsar la barra espaciadora.

Mapa de memoria y advertencias

En la figura 2 tenéis un mapa de memoria completo. Las especificaciones señaladas con interrogación comienzan en direcciones indeterminadas, y no tienen por qué estar en ese orden en la memoria, pero ése es el orden en que se colocan cuando usamos el programa monitor. Las especificaciones con un nombre de etiqueta entre paréntesis comienzan en la dirección almacenada en dicha etiqueta.

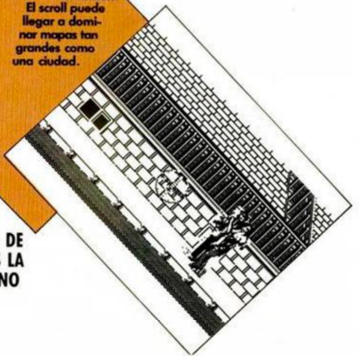
Los que tengan el número anterior, tal vez hayan advertido que el listado ensamblador 5 no es exactamente idéntico al de dicho número, sino que tiene algunas etiquetas más. Estas etiquetas señalan puntos en los que la nueva rutina de mapa hace modificaciones sobre la antigua rutina de sprites, sin embargo, no son indispensables para ensamblar, así que es como si los listados fueran idénticos. También puede que hayan notado que la subrutina de comprobación de choque entre sprites tiene distinta dirección de ejecución. Esto es porque ahora se inhabilitan las interrupciones durante su ejecución para poder usar el registro IY sin interferir con la rutina de interrupciones de la ROM.

Por último advertir que si definimos demasiados sprites o demasiado grandes tal vez no tengamos suficiente con el área reservada SPARES para almacenar trozos de pantalla. La única solución, como ya dijimos en su día, es cambiar la ubicación de la zona SPARES en el listado ensamblador, a no ser que podamos reducir la cantidad o tamaño de los sprites.





Impossaball. Conseguía con ayuda de tres scrolls una profundidad increíble.



Prohibition. El scroll puede llegar a dominar mapas tan grandes como una ciudad.

El sprite principal es necesario definirlo también dentro de la opción 2 del menú.

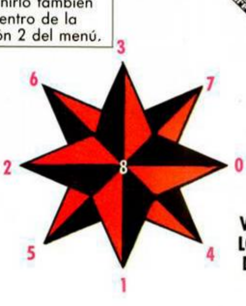


FIG. 1
VALOR PARA LAS DIRECCIONES DE LOS SCROLLS (LA DIRECCIÓN ES LA DE MOVIMIENTO DEL SPRITE, NO DEL MAPA).

LISTADO 1. CREADOR

```
  10 REM *********************
  20 REM *                   *
  30 REM * CREADOR DE DATOS  *
  40 REM *                   *
  50 REM *  PARA SPRITES Y   *
  60 REM *                   *
  70 REM *  PARA DECORADOS   *
  80 REM *                   *
  90 REM * POR PABLO ARIZA.  *
 100 REM *                   *
 110 REM *********************
 120 CLEAR 39999
 130 GO SUB 400: REM CARGAR
 140 GO SUB 600: REM INICIO
 150 GO SUB 900: REM LOAD GRAFS?
 160 GO SUB 1E3: REM FORMATOS
 170 GO SUB 6E3: REM MENU
 180 GO SUB 7E3: REM MAS?
 190 GO TO  140: REM REINICIO
 400
 410 REM *    CARGAR    *
 420
 430 PAPER 0: BORDER 0: INK 7
 440 CLS
 450 LOAD "CODSPRIT"CODE 61953,1
495
 460 LOAD "CODMAP"CODE 51900,992
 470 RESTORE 500: FOR X=0 TO 53
 480 READ A: POKE 63500+X,A
 490 NEXT X: RETURN
 500 DATA 33,0,0,17,0,0,1,0,0
 510 DATA 237,184,221,33,0,250
 520 DATA 1,0,0,221,110,0,221
 530 DATA 102,1,124,181,200,94
 540 DATA 35,86,235,9,235,114,43
 550 DATA 115,221,35,221,35,24
 560 DATA 232,33,0,0,17,0,0,1,0
 570 DATA 0,237,176,201
 600
 610 REM *    INICIO    *
 620
 630 LET NUMSPR=0: LET NUMPAI=0
 640 LET NUMAPS=0: LET NUGRAS=0
 650 LET TADECO   =51900
 660 LET TAMAPA   =51900
 670 LET TATRAYS  =51900
 680 LET DMAPX    =52970
 690 LET DMAPY    =52971
 700 LET DFILA    =52973
 710 LET DCOLUMN  =52972
 720 LET DWIDSC   =52974
 730 LET DHIGHSC  =52975
 740 LET DCALCUL  =52976
 750 LET DFILAPE  =52977
 760 LET DCOLUPE  =52978
 770 LET DIRPE0   =52979
 780 LET DVESCRO  =52995
 790 LET DTATABS  =52934
 800 LET DTADEFS  =52950
 810 LET DTADECO  =52931
 820 LET DMAIN    =52968
 830 LET DBACK    =52967
 840 LET DNUMDEC  =52933
 850 LET PUNTABL  =64000
 860 DIM C(8)
 870 POKE DNUMDEC,0
 880 POKE 23658,8
 890 RETURN
 900
 910 REM * CARGA GRAFS. *
 920
 930 LET A$="QUIERES CARGAR GRAF
ICOS": LET CU=0
 940 GO SUB 8E3: REM (S/N)?
 950 IF A$="N" THEN RETURN
 960 INPUT "CUANTO OCUPAN? ";CU:
 IF CU>51900-40000 THEN PRINT #0
; FLASH 1;"     NO CABEN EN LA M
EMORIA     ": BEEP .8,-12: PAUSE
 0: GO TO 930
 970 INPUT "DIRECCION? ";TUDIR
 980 INPUT "NOMBRE? "; LINE A$
 990 LOAD *"M";1;A$CODE 4E4: RET
URN
1000
1010 REM *   FORMATOS   *
1020
1030 CLS
1040 PRINT AT 21,0;"DATOS DEL MA
PA"
1050 LET A$="COLOR FONDO (+8*BRI
LLO)"
1060 LET MIN=0: LET MAX=15
1070 GO SUB 8100: REM INPUT
1080 POKE DBACK,DATO*8
1090 LET A$="ANCHO"
1100 LET MIN=1: LET MAX=255
1110 GO SUB 8100: REM INPUT
1120 POKE DMAPX,DATO
1130 LET A$="ALTO"
1140 LET MIN=1: LET MAX=255
1150 GO SUB 8100: REM INPUT
1160 POKE DMAPY,DATO
1170 LET A$="VELOC. DEL SCROLL"
1180 LET MIN=0: LET MAX=255
1190 GO SUB 8100: REM INPUT
1200 POKE DVESCRO,DATO
1210 GO SUB 8200: REM CORRECTO?
1220 IF A$="N" THEN GO TO 1E3
1230 CLS
1240 PRINT AT 21,0;"FORMATO DE L
A PANTALLA"
1250 LET A$="FILA INICIAL"
1260 LET MIN=0: LET MAX=23
1270 GO SUB 8100: REM INPUT
1280 POKE DFILA,DATO: LET F=DATO
1290 LET A$="COLUMNA INICIAL"
1300 LET MIN=0: LET MAX=31
1310 GO SUB 8100: REM INPUT
1320 POKE DCOLUMN,DATO
1330 LET A$="ANCHO (CARACTERES)"
1340 LET MIN=1: LET MAX=32-DATO
1350 GO SUB 8100: REM INPUT
1360 POKE DWIDSC,DATO: LET WI=DA
TO
1370 LET A$="ALTO (CARACTERES)"
1380 LET MIN=1: LET MAX=24-F
1390 GO SUB 8100: REM INPUT
1400 POKE DHIGHSC,DATO: POKE DCA
LCUL,DATO+INT (WI*DATO/256)
1410 GO SUB 8200: REM CORRECTO?
1420 IF A$="N" THEN GO TO 1230
1430 CLS
1440 PRINT AT 21,0;"DATOS DEL SP
RITE PRINCIPAL"
1450 LET A$="COORD X (PIXELS)"
1460 LET MIN=0: LET MAX=255
1470 GO SUB 8100: REM INPUT
1480 POKE DCOLUPE,DATO
1490 LET A$="COORD Y (PIXELS)"
1500 LET MIN=0: LET MAX=191
1510 GO SUB 8100: REM INPUT
1520 POKE DFILAPE,DATO
1530 LET A$="NUMERO"
1540 LET MIN=0: LET MAX=16
1550 GO SUB 8100: REM INPUT
1560 POKE DMAIN,DATO
1570 RESTORE 1700
1580 LET PUNTERO=DIRPE0
1590 FOR X=0 TO 7: READ A$
1600 LET A$="DIR DE GRAFS "+A$
1610 LET MIN=0: LET MAX=65535
1620 GO SUB 8100: REM INPUT
1630 LET OFFSET=2*X
1640 GO SUB 8300: REM POKE DOBLE
1650 IF DATO=0 THEN LET X=7
1660 NEXT X
1670 GO SUB 8200: REM CORRECTO?
1680 IF A$="N" THEN GO TO 1430
1690 RETURN
1700 DATA "DERECHA","ABAJO"
1710 DATA "IZQUIERDA","ARRIBA"
1720 DATA "AB. DER.","AB. IZQ."
1730 DATA "AR. IZQ.","AR. DER."
2000
2010 REM * DEF. SPRITES *
2020
2030 CLS : PRINT TAB 4;"DEFINICI
ON DE SPRITES": LET ACTU=0
2040 IF NUMSPR=0 THEN GO TO 2100
2050 PRINT AT 21,0;"ULTIMO SPRIT
E DEFINIDO: ";NUMSPR-1
2060 LET S=NUMSPR: LET A$="QUE S
PRITE (S=SIGUIENTE)"
2070 LET MIN=0: LET MAX=S
2080 GO SUB 8100: REM INPUT
2090 LET ACTU=DATO
2100 CLS : PRINT AT 21,0;"SPRITE
 NUMERO ";ACTU: LET PUNTERO=5300
0+17*ACTU
2110 LET A$="ANCHO (CARACTERES)"
2120 LET MIN=0: LET MAX=32
2130 GO SUB 8100: REM INPUT
2140 IF DATO=0 THEN RETURN
2150 POKE PUNTERO+2,DATO: LET CO
=DATO
2160 LET A$="ALTO (PIXELS)"
2170 LET MIN=1: LET MAX=192
2180 GO SUB 8100: REM INPUT
2190 POKE PUNTERO+3,DATO: LET FI
=DATO
2200 LET A$="DIR. DEL GRAFICO"
2210 LET MIN=1: LET MAX=65535
2220 GO SUB 8100: REM INPUT
2230 LET OFFSET=0
2240 GO SUB 8300: REM POKE DOBLE
2250 LET A$="FASES DE ANIMACION"
2260 LET MIN=1: LET MAX=255
2270 GO SUB 8100: REM INPUT
2280 POKE PUNTERO+8,DATO
2290 IF DATO<=2 THEN LET F1=0: G
O TO 2340
2300 LET A$="0:CICLICA 1:ADELANT
E/ATRAS"
2310 LET MIN=0: LET MAX=1
2320 GO SUB 8100: REM INPUT
2330 LET F1=DATO
2340 LET A$="COORD X INICIAL"
2350 LET MIN=-32768: LET MAX=327
67
2360 GO SUB 8100: REM INPUT
2370 LET OFFSET=4
2380 GO SUB 8300: REM POKE DOBLE
2390 LET A$="COORD Y INICIAL"
2400 LET MIN=-32768: LET MAX=327
67
2410 GO SUB 8100: REM INPUT
2420 LET OFFSET=6
2430 GO SUB 8300: REM POKE DOBLE
2440 LET A$="COLOR (+8*BRILLO)"
2450 LET MIN=0: LET MAX=15
2460 GO SUB 8100: REM INPUT
2470 LET F1=F1*8+DATO+56*(DATO>7
)
2480 LET A$="SE MUEVE EN LINEA R
ECTA"
2490 GO SUB 8000: REM (S/N)?
2500 POKE PUNTERO+9,F1+16*(A$="N
")
2510 LET DATO=FI*CO
2520 LET OFFSET=15
2530 GO SUB 8300: REM POKE DOBLE
2540 IF A$="N" THEN GO TO 2690
2550 LET A$="INCREMENTO X"
2560 LET MIN=-128: LET MAX=+127
2570 GO SUB 8100: REM INPUT
2580 POKE PUNTERO+10,DATO
2590 LET A$="INCREMENTO Y"
2600 LET MIN=-128: LET MAX=+127
2610 GO SUB 8100: REM INPUT
2620 POKE PUNTERO+11,DATO
2630 GO SUB 8200: REM CORRECTO?
2640 IF A$="N" THEN GO TO 2100
2650 IF ACTU<>NUMSPR THEN RETURN
2660 LET ACTU=ACTU+1
2670 LET NUMSPR=ACTU
2680 GO TO 2100
2690 LET A$="DEFINIR LA TRAYECTO
RIA"
2700 GO SUB 8000: REM (S/N)?
2710 IF A$="S" THEN GO TO 2800
2720 LET A$="DIR. DE LA TRAYECTO
RIA"
2730 LET MIN=0: LET MAX=655365
2740 GO SUB 8100: REM INPUT
2750 LET OFFSET=10
2760 GO SUB 8300: REM POKE DOBLE
2770 LET OFFSET=13
2780 GO SUB 8300: REM POKE DOBLE
2790 GO TO 2630
2800 REM DEFINICION TRAYECTORIA
2810 LET T$=""
2820 LET A$="INC X (O DIR GRAF)"
2830 LET MIN=-128: LET MAX=65535
2840 GO SUB 8500: REM INPUT2
2850 IF DATO<128 THEN GO TO 2890
2860 RANDOMIZE DATO
2870 LET T$=T$+CHR$ 126+CHR$ PEE
```



```
K 23670+CHR$ PEEK 23671
2880 GO TO 2820
2890 LET T$=T$+CHR$ DATO
2900 IF DATO=127 THEN GO TO 2960
2910 LET A$="INC Y"
2920 LET MIN=-128: LET MAX=+127
2930 GO SUB 8500: REM INPUT2
2940 LET T$=T$+CHR$ DATO
2950 GO TO 2820
2960 GO SUB 9000: REM RESTO
2970 IF A$="N" THEN GO TO 2480
2980 GO TO 2750
3000
3010 REM *  DECORADOS.  *
3020
3030 CLS : PRINT TAB 4;"DEFINICI
ON DE DECORADOS": LET ACTU=0
3040 IF NUGRAS=0 THEN GO TO 3100
3050 PRINT AT 21,0;"ULTIMO DECOR
ADO DEFINIDO: ";NUGRAS-1
3060 LET S=NUGRAS: LET A$="QUE D
ECORADO (S=SIGUIENTE)"
3070 LET MIN=0: LET MAX=S
3080 GO SUB 8100: REM INPUT
3090 LET ACTU=DATO: IF ACTU<NUGR
AS THEN LET PUNTERO=TADECO+5*ACT
U: GO TO 3140
3100 LET SPAC=5
3110 GO SUB 8400: REM ABRE SPAC
3120 IF ERROR=1 THEN RETURN
3130 LET PUNTERO=TATRAYS: LET TA
TRAYS=ANTIGUO
3140 CLS : PRINT AT 21,0;"DECORA
DO NUMERO ";ACTU
3150 LET A$="ANCHO (CARACTERES)"
3160 LET MIN=1: LET MAX=32
3170 GO SUB 8100: REM INPUT
3180 POKE PUNTERO+2,DATO
3190 LET A$="ALTO (PIXELS)"
3200 LET MIN=1: LET MAX=192
3210 GO SUB 8100: REM INPUT
3220 POKE PUNTERO+3,DATO
3230 LET A$="COLOR (+8*BRILLO)"
3240 LET MIN=0: LET MAX=15
3250 GO SUB 8100: REM INPUT
3260 POKE PUNTERO+4,DATO+56*(DAT
O>7)+PEEK DBACK-64*(PEEK DBACK>6
3 AND DATO>7)
3270 LET A$="DIR DEL GRAFICO"
3280 LET MIN=0: LET MAX=65535
3290 GO SUB 8100: REM INPUT
3300 LET OFFSET=0
3310 GO SUB 8300: REM POKE DOBLE
3320 GO SUB 8200: REM CORRECTO?
3330 IF A$="N" THEN GO TO 3140
3340 IF ACTU<NUGRAS THEN RETURN
3350 LET ACTU=ACTU+1
3360 LET NUGRAS=ACTU
3370 LET A$="DEFINIR MAS DECORAD
OS"
3380 GO SUB 8000: REM (S/N)?
3390 IF A$="N" THEN RETURN
3400 GO TO 3100
4000
4010 REM *   DEF MAPA   *
4020
4030 CLS : PRINT TAB 4;"DEFINICI
ON DE MAPA": LET ACTU=0
4040 IF NUMPAI=0 THEN GO TO 4100
4050 PRINT AT 21,0;"ULTIMO ELEME
NTO DEFINIDO: ";NUMPAI-1
4060 LET S=NUMPAI: LET A$="QUE E
LEMENTO (S=SIGUIENTE)"
4070 LET MIN=0: LET MAX=S
4080 GO SUB 8100: REM INPUT
4090 LET ACTU=DATO: IF ACTU<NUMP
AI THEN LET PUNTERO=TAMAPA+3*ACT
U: GO TO 4140
4100 LET SPAC=3
4110 GO SUB 8700: REM ABRE SPAC2
4120 IF ERROR=1 THEN RETURN
4130 LET PUNTERO=TADECO-3
4140 CLS : PRINT AT 21,0;"ELEMEN
TO NUMERO ";ACTU
4150 LET A$="CODIGO DE DECORADO"
4160 LET MIN=0: LET MAX=NUGRAS-1
4170 GO SUB 8100: REM INPUT
4180 POKE PUNTERO,DATO
4190 LET A$="COORD X (CARACTERES
)"
4200 LET MIN=0: LET MAX=255
4210 GO SUB 8100: REM INPUT
4220 POKE PUNTERO+1,DATO
4230 LET A$="COORD Y (CARACTERES
)"
4240 LET MIN=0: LET MAX=255
4250 GO SUB 8100: REM INPUT
4260 POKE PUNTERO+2,DATO
4270 GO SUB 8200: REM CORRECTO?
4280 IF A$="N" THEN GO TO 4140
4290 IF ACTU<NUMPAI THEN RETURN
4300 LET ACTU=ACTU+1
4310 LET NUMPAI=ACTU
4320 LET A$="DEFINIR MAS ELEMENT
OS"
4330 GO SUB 8000: REM (S/N)?
4340 IF A$="N" THEN RETURN
4350 GO TO 4100
5000
5010 REM * GRABAR TODO. *
5020
5030 CLS : PRINT "ESPERA UN POCO
": GO SUB 5500
5040 LET PUNTERO=DTATABS: LET DI
R=TAMAPA
5050 FOR X=1 TO 8
5060 LET OFFSET=2*X+14
5070 LET DATO=C(X)
5080 GO SUB 8300: REM POKE DOBLE
5090 LET OFFSET=2*X-2
5100 LET DATO=DIR
5110 GO SUB 8300: REM POKE DOBLE
5120 LET DIR=DIR+C(X)*3
5130 NEXT X: RANDOMIZE TADECO
5140 LET START=TAMAPA: POKE DTAD
ECO,PEEK 23670: POKE DTADECO+1,P
EEK 23671: POKE DNUMDEC,NUGRAS
5150 IF NOT CU THEN GO TO 5440
5160 RANDOMIZE CU
5170 POKE 63507,PEEK 23670: POKE
 63508,PEEK 23671
5180 RANDOMIZE 40000+CU-1
5190 POKE 63501,PEEK 23670: POKE
 63502,PEEK 23671
5200 RANDOMIZE TAMAPA-1
5210 POKE 63504,PEEK 23670: POKE
 63505,PEEK 23671
5220 RANDOMIZE TAMAPA-(TUDIR+CU)
5230 POKE 63516,PEEK 23670: POKE
 63517,PEEK 23671
5240 LET PUNTERO=PUNTABL: LET OF
FSET=0
5250 FOR X=TADECO TO TATRAYS-5 S
TEP 5
5260 LET DATO=X
5270 GO SUB 8300: REM POKE DOBLE
5280 LET OFFSET=OFFSET+2
5290 NEXT X
5300 FOR X=53000 TO 53000+17*(NU
MSPR-1) STEP 17
5310 LET DATO=X
5320 GO SUB 8300: REM POKE DOBLE
5330 LET OFFSET=OFFSET+2
5340 NEXT X
5350 FOR X=DIRPE0 TO DIRPE0+14 S
TEP 2
5360 LET DATO=X
5370 GO SUB 8300: REM POKE DOBLE
5380 LET OFFSET=OFFSET+2
5390 NEXT X
5400 LET DATO=0
5410 GO SUB 8300: REM POKE DOBLE
5420 RANDOMIZE USR 63500
5430 LET START=TAMAPA-CU
5440 GO SUB 5450: GO SUB 7500: R
ETURN
5450 CLS : LET LONG=53289-START
5460 INPUT "NOMBRE? "; LINE A$
5470 IF A$="" OR LEN A$>9 THEN B
EEP .8,-12: GO TO 5450
5480 PRINT A$;"1 ";START;",";LON
G: SAVE A$+"1"CODE START,LONG
5490 PRINT A$;"2 ";61953;",";149
5: SAVE A$+"2"CODE 61953,1495
5500
5510 REM *  NUEVO MAPA  *
5520
5530 IF NUMAPS=0 THEN LET C(1)=N
UMPAI: LET NUMAPS=1: LET RESTO=N
UMPAI: RETURN
5540 IF NUMAPS=8 THEN LET C(8)=C
(8)+NUMPAI-RESTO: GO TO 5570
5550 LET NUMAPS=NUMAPS+1
5560 LET C(NUMAPS)=NUMPAI-RESTO
5570 LET RESTO=NUMPAI
5580 RETURN
6010 REM *     MENU     *
6020
6030 RESTORE 6130: CLS
6040 PRINT "   DEFINICION DE MAP
A Y SPRITES"''
6050 PRINT TAB 14;"MENU"''
6060 FOR X=1 TO 6: READ A$
6070 PRINT ''TAB 8;X;".-";A$;
6080 NEXT X
6090 LET A$=INKEY$
6100 IF A$>"7" OR A$<"1" THEN GO
 TO 6090
6110 LET OPCION=VAL A$: GO SUB 1
000*OPCION-500*(OPCION=6): IF OP
CION=5 THEN RETURN
6120 GO TO 6000
6130 DATA "REPETIR FORMATOS"
6140 DATA "DEFINIR SPRITES"
6150 DATA "DEFINIR DECORADOS"
6160 DATA "DEFINIR MAPA"
6170 DATA "GRABARLO TODO"
6180 DATA "NUEVO MAPA"
7000
7010 REM *   OTRA VEZ?  *
7020
7030 LET A$="QUIERES RECOMENZAR"
7040 GO SUB 8000: REM (S/N)?
7050 IF A$="S" THEN RETURN
7060 CLEAR START-1: NEW
7500
7510 REM *  GRABAR MAS  *
7520 CLS : PRINT "PULSA ENTER PA
RA VOLVER A GRABAR"''"O CUALQUIE
R TECLA PARA FINALIZAR"
7530 PAUSE 0: IF CODE INKEY$=13
THEN GO SUB 5480: GO TO 7520
7540 RETURN
8000
8010 REM *    (S/N)?    *
8020
8030 PRINT #0;A$;" (S/N)? "
8040 LET A$=INKEY$
8050 IF A$<>"S" AND A$<>"N" THEN
 GO TO 8040
8060 INPUT "": RETURN
8100
8110 REM * INPUT NUMERO *
8120
8130 INPUT (A$+"? ");DATO
8140 IF DATO>MAX OR DATO<MIN THE
N BEEP .8,-12: GO TO 8130
8150 PRINT A$;": ";DATO
8160 RETURN
8200
8210 REM *  CORRECTO?  *
8220
8230 LET A$="DATOS CORRECTOS"
8240 GO TO 8000: REM (S/N)?
8300
8310 REM *  POKE DOBLE  *
8320
8330 LET HIGH=INT (DATO/256)
8340 LET LOW=DATO-256*HIGH
8350 POKE PUNTERO+OFFSET,LOW
8360 POKE PUNTERO+OFFSET+1,HIGH
8370 RETURN
8400
8410 REM *  ABRE  SPAC  *
8420
8430 IF TAMAPA-CU-SPAC<40000 THE
N PRINT #0; FLASH 1;"  ERROR: NO
 CABE EN LA MEMORIA  ": BEEP 1,-
12: PAUSE 0: LET ERROR=1: RETURN
8440 LET ANTIGUO=TATRAYS
8450 LET TATRAYS=TATRAYS-SPAC
8460 LET ORI=TAMAPA: LET BYT=ANT
IGUO-TAMAPA
8470 LET TADECO=TADECO-SPAC
8480 LET TAMAPA=TAMAPA-SPAC
8490 LET DES=TAMAPA: GO TO 8600:
 REM MOVER
8500
8510 REM *   INPUT  2   *
8520
8530 INPUT (A$+"? ");DATO
8540 IF DATO>MAX OR DATO<MIN THE
N BEEP .8,-12: GO TO 8130
8550 RETURN
8600
8610 REM *    MOVER.    *
8620
8630 RANDOMIZE ORI
8640 POKE 63543,PEEK 23670: POKE
 63544,PEEK 23671
8650 RANDOMIZE DES
8660 POKE 63546,PEEK 23670: POKE
 63547,PEEK 23671
8670 RANDOMIZE BYT
8680 POKE 63549,PEEK 23670: POKE
 63550,PEEK 23671: IF BYT THEN R
ANDOMIZE USR 63542
8690 LET ERROR=0: RETURN
8700
8710 REM *  ABRE SPAC2  *
8720
8730 IF TAMAPA-CU-SPAC<40000 THE
N PRINT #0; FLASH 1;"  ERROR: NO
 CABE EN LA MEMORIA  ": BEEP 1,-
12: PAUSE 0: LET ERROR=1: RETURN
8740 LET ORI=TAMAPA: LET BYT=TAD
ECO-TAMAPA
8750 LET TAMAPA=TAMAPA-SPAC
8760 LET DES=TAMAPA: GO TO 8600:
 REM MOVER
9000
9010 REM * TRAYECTORIA. *
9020
9030 LET A$="TRAYECTORIA CORRECT
A": GO SUB 8000: REM (S/N)?
9040 IF A$="N" THEN RETURN
9050 LET SPAC=LEN T$
9060 GO SUB 8400: REM ABRE SPAC
9070 IF ERROR=1 THEN LET A$="N":
 RETURN
9080 LET TEMPO=PUNTERO: FOR X=1
TO LEN T$
9090 POKE TATRAYS+X-1,CODE T$(X)
9100 IF CODE T$(X)=126 AND X/2<>
INT (X/2) THEN LET DATO=TATRAYS+
X: LET OFFSET=0: LET PUNTERO=PUN
TABL: LET PUNTABL=PUNTABL+2: GO
SUB 8300: REM POKE DOBLE
9110 NEXT X: LET PUNTERO=TEMPO:
LET DATO=TATRAYS
9120 PRINT AT 0,0;"LA TABLA DE T
RAYECTORIA HA SIDO COLOCADA EN L
A DIRECCION ";TATRAYS''"APUNTALO
```

LISTADO 2. CODSPRIT

Línea	Datos	Control
1	F33EF1ED4706002100F1	1134
2	36F22310FB36F21128D5	1164
3	2100403EC008E5012000	621
4	EDB013E1247CE607200A	1096
5	7DC6206F38047CD60867	975
6	083D20E321005811E8ED	935
7	3E18012000EDB0133D20	644
8	F7ED5E0E0426F83E0991	1098
9	912E00451E0057152806	444
10	CB38CB1B18F770247325	1060
11	2D20EC24240D20E13E01	718
12	329DF7AF32815C329CF7	1353
13	FBC93E3FED56ED47C9ED	1646
14	4BB05C0C280A0DCDB3F2	1044
15	010000D003C9DD21805C	935
16	3A815CDD340008ED4BB0	1048
17	5C78B92807CDB3F20101	1072
18	00D8083D20E90BC921B9	980
19	F722B7F7219EF722EAF2	1659
20	CDCFF2D021C0F722B7F7	1798
21	21AAF722EAF278CDB3F7	1711
22	EB79CDB3F7A7ED527830	1641
23	087C2F677D2F6F2379CD	926
24	C7F7C300000000000000	641
25	00F3F5C5D5E5DDE5D908	1802
26	F5C5D5E5FDE53A9CF7A7	1994
27	C213F3CDAFF33E01329C	1348
28	F7C39EF32128D1AF329C	1506
29	F73A9DF7DD2129D03D28	1313
30	4A08DD5E00DD5601DD7E	1052
31	0E3244F3325AF3ED44C6	1261
32	213248F3325EF3DD7E06	1138
33	06000E00EDB00E00EB09	691
34	EB3DC243F3DD7E0DDD5E	1475
35	0BDD560C0E00EDB00E00	771
36	EB09EB3DC259F30E0FDD	1316
37	0908C321F33E01329DF7	1005
38	1128D1DD2129D0FD2108	1063
39	CF3A815C47A72814C5D5	1194
40	CD5FF4D1CD1BF5011100	1248
41	FD09C110EFCDE5F6FDE1	1868
42	FDE5FFFDE1E1D1C1F108	2091
43	D9DDE1E1D1C1F1FBED4D	2096
44	2128D51100400E00D921	631
45	E8ED1100580100190906	823
46	09EDA0EDA0EDA0EDA0ED	1834
47	A0EDA0EDA0EDA0EDA0ED	1985
48	A0EDA0EDA0EDA0EDA0ED	1985
49	A0EDA0EDA0EDA0EDA0ED	1985
50	A0EDA0EDA0EDA0EDA0ED	1985
51	A0EDA0EDA0EDA0EDA0ED	1985
52	A0EDA0EDA07BD6205F38	1474
53	0114231086 7BC6205F38	758
54	047AD60857D9EDA0EDA0	1446
55	EDA0EDA0EDA0EDA0EDA0	1985
56	EDA0EDA0EDA0EDA0EDA0	1985
57	EDA0EDA0EDA0EDA0EDA0	1985
58	EDA0EDA0EDA0EDA0EDA0	1985
59	EDA0EDA0EDA0EDA0EDA0	1985
60	EDA0EDA0EDA0EDA0EDA0	1985
61	2305C2C1F3C9FD7E05FE	1509
62	7FC8FD7E0957E647DD77	1443
63	0ACB622831FD6E0DFD66	1131
64	0E7EFE7F2007FD6E0AFD	1186
65	660B7EFE7E200E237EFD	1079
66	7700237EFD770123C378	1003
67	F45F237E23FD750DFD74	1287
68	0EC3AAF4FD7E0A5FFD7E	1486
69	0B4FFE803F9F47FD6E06	1134
70	FD660709FD7506FD7407	1123
71	7B4FFE803F9F47FD6E04	1244
72	FD660509FD7504FD7405	1117
73	FD7E0CFD5E08CB5AC2E7	1464
74	F43CBB2001AFFD770CC3	1278
75	FDF43C579393C60232FA	1438
76	F428017AFD770CBB3804	1038
77	3E00ED44FD5E0FFD5610	1084
78	CB23CB12FD6E00FD6601	1178
79	A72804471910FDDD7502	916
80	DD7403C9FD7E044FE606	1239
81	C6F8DD7709FD7E05A728	1386
82	4D3CC079C606301321E7	985
83	ED0127D5FD7E02DD7707	1218
84	AFDD7708C3C1F5371FCB	1445
85	2FCB2F47FD8602D0C8DD	1386
86	770778ED44DD7708DD86	1254
87	02DD77023E00DD8E03DD	993
88	77037921E7ED0127D5E6	1227
89	06204D2303C3C1F579D6	1121
90	013836CB3FCB3FCB3FFD	1162
91	8602FE213829D620DD77	1106
92	08ED44FD8602DD770779	1170
93	CB3FCB3FCB3F4F21E8ED	1379
94	856F8C9567790128D581	1140
95	4F889147C3C1F5FD7E02	1445
96	DD7707AFDD7708C398F5	1462
97	FD7E07A72838 3CC0FD7E	1283
98	06FD8603D0C8DD7706ED	1387
99	44FD8603E5D5DD6E02DD	1454
100	66031600FD5E02193DC2	756
101	E6F5DD7502DD7403DD7E	1502
102	06C607CB3FCB3FCB3FDD	1230
103	770DC376F6FD7E06FEC0	1522
104	D0CB3FCB3FCB3FC50121	1237
105	003C3D280409C313F6C1	827
106	E5D56069112100FD7E06	1078
107	A72807193DC228F6444D	925
108	FD7E06FD8603382CFEC1	1322
109	3028FD7E03DD77067908	945
110	FD4E06CB39CB39CB39FD	1370
111	7E06FD8603C607CB3FCB	1196
112	3FCB3F91DD770D084FC3	1109
113	76F63EC0FD9606DD7706	1373
114	C607CB3FCB3FCB3FDD77	1343
115	0DDD7100DD7001DD6E02	1014
116	DD6603FD5E0FFD561019	1068
117	DD7504DD7405DD5607DD	1219
118	7E09E606280114DD720E	781
119	D1E1E5DD750BDD740C60	1457
120	69DD7E0E32C8F632D6F6	1472
121	ED44C62132CCF632DAF6	1550
122	3A9DF73C329DF7DD7E06	1329
123	06000E00EDB00E00093D	517
124	20F6E1DD7E0D0E00EDB0	1290
125	0E00093DC2D5F60E0FDD	987
126	09C9DD2129D03A9DF73D	1236
127	C8F5DD5E00DD5601DD66	1391
128	092EFF7E3255F7247E25	1017
129	322BF7D9DD6E02DD6603	1216
130	E5FDE1DD6E04DD6605DD	1591
131	7E07322FF7ED44C62132	1063
132	5BF7DD4E06DD5E081600	988
133	D9010000D90600FD7E00	820
134	FD23087E23D96F7EB124	1124
135	4E086F784625B66F08EB	960
136	A6B37723EBD910E119FD	1470
137	19D979F600EBA6B0773E	1367
138	00856F8C9567EBD90DC2	1295
139	29F7DD6E0BDD660CDD7E	1312
140	0E3284F7ED44C621328D	1170
141	F7DD5E0A1688DD7E0D08	1146
142	06007EA2B3772310F90E	906
143	0009083DC282F7F10E0F	919
144	DD09C3ECF60001FD5E02	1257
145	CB23CB23CB23C3ADF7FD	1582
146	5E031600A7ED52C9CDC7	1210
147	F7C3006CFD6E04FD6605	1169
148	C9FD6E06FD6607C9FD21	1419
149	08CFA7C8D5111100FD19	1107
150	3D20FBD1C906142C2C14	888

DUMP: 40.000. N.° BYTES: 1495

LISTADO 3. CODMAP

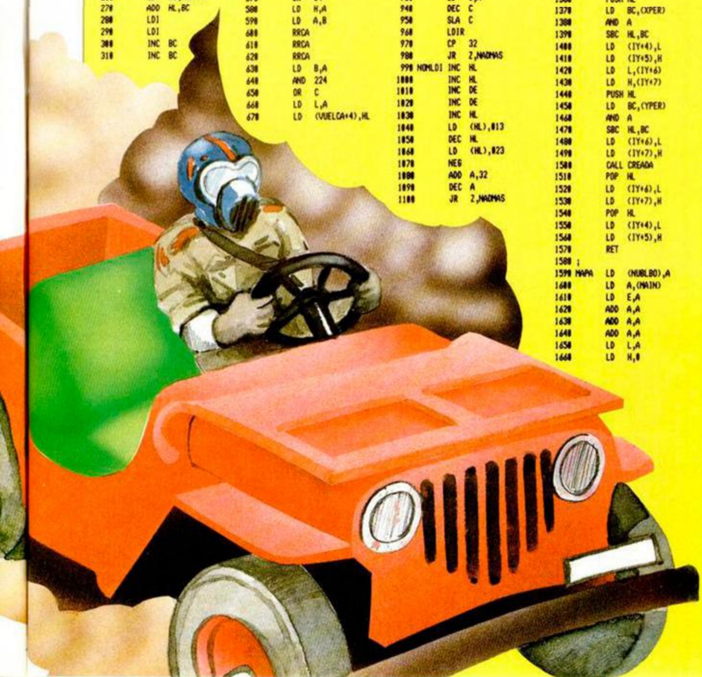
Línea	Datos	Control
1	21004011014001001875	321
2	EDB001FF023AE7CE77ED	1522
3	B021C6CE3AE6CE874F06	1327
4	00118FCE09EDA0EDA003	1220
5	0321D6CE09EDA0EDA03E	1321
6	CD3270F321DECB2271F3	1458
7	21A4CB228CF31128D52A	1129
8	ECCE4D4419EB26006829	1030
9	2929192280F378E618F6	1180
10	4067780F0F0F47E6E0B1	1034
11	6F22B3F378E61F5FC658	1329
12	67229BCB6311E8ED19EB	1340
13	2AEDCE26001922980CB3A	995
14	F0CE329FCB11C6F321C5	1546
15	F336A02B36ED3AEECE06	1299
16	00FE01280A4F0DCB21ED	870
17	B0FE2028152323131323	666
18	36132B3623ED44C6203D	801
19	2804874FEDB021C4F311	1160
20	19F4014000EDB02195CB	1132
21	11B5F3010F00EDB021C0	1095
22	F3225CF4C301F200D921	1301
23	00001100001FF00D901	491
24	0009FD6E04FD6605E5ED	1202
25	4B04CFA7ED42FD7504FD	1383
26	7405FD6E06FD6607E5ED	1318
27	4B06CFA7ED42FD7506FD	1387
28	7407CD18F5E1FD7506FD	1454
29	7407E1FD7504FD7405C9	1297
30	329DF73AE8CE5F878787	1450
31	6F26005444 2919FD2108	661
32	CFE8FD193AE9CEFE0828	1519
33	43874F21F3CE095E2356	987
34	FD7300FD72013A03CFA7	1171
35	282E218CCD097E4F179F	908
36	47237E5F179F57FD6E04	963
37	FD66053A03CF093D20FC	982
38	4D44FD6E06FD66073A03	937
39	CF193D20FCEB180CFD4E	1179
40	04FD4605FD5E06FD5607	1031
41	78173003010000 7A1730	388
42	031100002AEACE260029	581
43	2929A7ED423003 09444D	757
44	2AEBCE2600292929A7ED	1048
45	52300219EBFD7104FD70	1127
46	05FD7306FD72073AF2CE	1259
47	3CED446F26FF23092204	851
48	CF3AF1CE3CED446F26FF	1481
49	23192206CF2AC1CE7CB5	1053
50	C83EC3329FF61148CDED	1443
51	53A0F6DD2ABFCEE5DD7E	1725
52	003C2872DD6E01260029	625
53	2929ED5B04CFA7ED527C	1231
54	A728043CC230CD229DCE	1115
55	DD6E022600292929ED5B	822
56	06CFA7ED527CA728043C	1094
57	C230CD229FCEDD5E0016	1183
58	00FD2AC3CEFD19EB2929	1291
59	EBFD19FD7E0032AACEFD	1571
60	7E0132ABCEFD7E02329B	1140
61	CEFD7E03329CCEFD7E04	1383
62	3259CDDDE5DD21A8CEFD	1675
63	2199CECD18F5DDE10103	1319
64	00DD09E12B7CB5C2B7CC	1384
65	3EE5329FF621DD7522A0	1311
66	F6C9DD4E0DDD7E0E325B	1261
67	CDED44C6215F16003E00	920
68	0600772310FC190D20F6	744
69	DD66092EFF7E32AECD24	1224
70	7E3298CD25DD7E07329C	1130
71	CDED44C62132B3CDDD5E	1490
72	00DD5601D9DD6E02DD66	1181
73	03DD5E081600DD4E05D9	870
74	1AE6004FD906007ED96F	1012
75	7EB11213244E25D92310	759
76	F2D91AE600B112EB0E00	1159
77	09EBD9190D20DAC90100	951
78	0001FF0000FF0101FF01	769
79	FFFF01FFF3CD82F2FBC9	2038
80	F33AB15C6F2600444D29	905
81	29292909010 8CF09E5FD	839
82	E12115CE3AB05C3219CE	1092
83	3C20032 11DCE2202CEDD	826
84	2ABFCEED4BC1CECD0000	1355
85	280D110300DD19 0B78B1	627
86	20F101FFFFFBC9DD7E00	1583
87	FE002801C9DD6E002600	865
88	C5444D292909ED4BC3CE	1146
89	09232344 4DDD6E01FD5E	903
90	04FD5605AF3281CE3291	1103
91	CECD70CE202603DD6E02	1135
92	FD5E06FD5607FD233E03	1052
93	3281CE3291CECD70CEFD	1562
94	28200BC12AC1CEA7ED42	1190
95	444DAFC9C1AF3CC92600	1188
96	292929A7ED52380F7CA7	971
97	C0FD7E021803878787BD	1194
98	D8AFC924C07DED446F0A	1371
99	1803878787BDD8AFC900	1213

DUMP: 51.900
N.° BYTES: 989

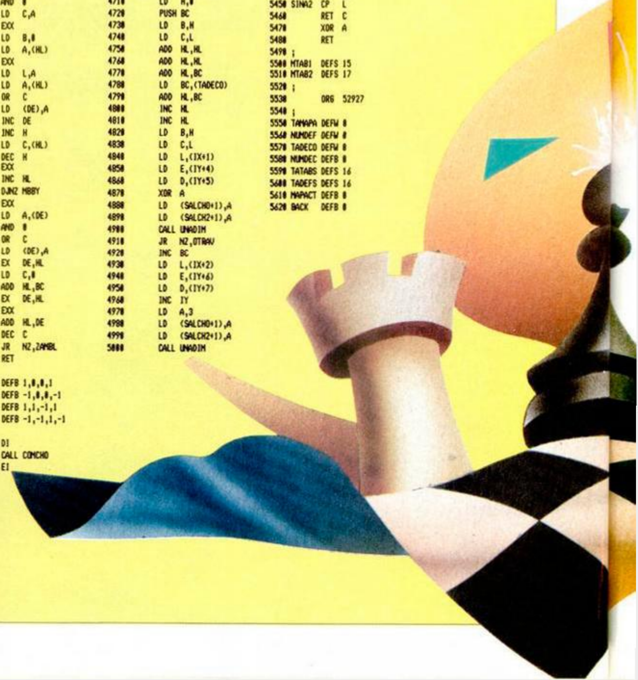


LISTADO 4. ENSMAP

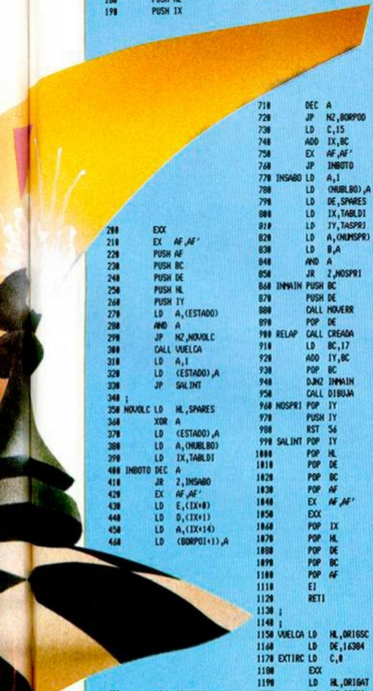
```
10 *D+
20 *C-
30 ;
40 ; CONTROLADOR DE MAPAS
50 ;
60 ;      CON SCROLL
70 ;
80 ; POR PABLO ARIZA 1987
90 ;
100        ORG  51900
110 ;
120 INIMAP LD   HL,16384
130        LD   DE,16385
140        LD   BC,6144
150        LD   (HL),L
160        LDIR
170        LD   BC,767
180        LD   A,(BACK)
190        LD   (HL),A
200        LDIR
210        LD   HL,TATABS
220        LD   A,(MAPACT)
230        ADD  A,A
240        LD   C,A
250        LD   B,0
260        LD   DE,TAMAPA
270        ADD  HL,BC
280        LDI
290        LDI
300        INC  BC
310        INC  BC
320        LD   HL,TADEFS
330        ADD  HL,BC
340        LDI
350        LDI
360        LD   A,#CD
370        LD   (INSABO+2),A
380        LD   HL,MAPA
390        LD   (INSABO+3),HL
400        LD   HL,RELASU
410        LD   (RELAP+1),HL
420        LD   DE,ORIGSC
430        LD   HL,(COLUMN)
440        LD   C,L
450        LD   B,H
460        ADD  HL,DE
470        EX   DE,HL
480        LD   H,0
490        LD   L,B
500        ADD  HL,HL
510        ADD  HL,HL
520        ADD  HL,HL
530        ADD  HL,DE
540        LD   (VUELCA+1),HL
550        LD   A,B
560        AND  24
570        OR   64
580        LD   H,A
590        LD   A,B
600        RRCA
610        RRCA
620        RRCA
630        LD   B,A
640        AND  224
650        OR   C
660        LD   L,A
670        LD   (VUELCA+4),HL
680        LD   A,H
690        AND  31
700        LD   E,A
710        ADD  A,88
720        LD   H,A
730        LD   (ATSET+4),HL
740        LD   H,E
750        LD   DE,ORIGAT
760        ADD  HL,DE
770        EX   DE,HL
780        LD   HL,(FILA)
790        LD   H,0
800        ADD  HL,DE
810        LD   (ATSET+1),HL
820        LD   A,(CALCUL)
830        LD   (ATSET+8),A
840        LD   DE,LOPPNU+2
850        LD   HL,LOPPNU+1
860        LD   (HL),#A0
870        DEC  HL
880        LD   (HL),#ED
890        LD   A,(WIDSC)
900        LD   B,0
910        CP   1
920        JR   Z,NOMLDI
930        LD   C,A
940        DEC  C
950        SLA  C
960        LDIR
970        CP   32
980        JR   Z,NADMAS
990 NOMLDI INC  HL
1000       INC  HL
1010       INC  DE
1020       INC  DE
1030       INC  HL
1040       LD   (HL),#13
1050       DEC  HL
1060       LD   (HL),#23
1070       NEG
1080       ADD  A,32
1090       DEC  A
1100       JR   Z,NADMAS
1110       ADD  A,A
1120       LD   C,A
1130       LDIR
1140 NADMAS LD  HL,LOPPNU
1150       LD   DE,SITERC+1
1160       LD   BC,#0040
1170       LDIR
1180       LD   HL,NUCOD
1190       LD   DE,EXTIRC
1200       LD   BC,15
1210       LDIR
1220       LD   HL,LVOLCA-1
1230       LD   (SALTO+1),HL
1240       JP   INICIO
1250 ;
1260 NUCOD NOP
1270       EXX
1280 ATSET LD   HL,0
1290       LD   DE,0
1300       LD   BC,#00FF
1310 NLVOLC EXX
1320       LD   BC,#0900
1330 ;
1340 RELASU LD  L,(IY+4)
1350       LD   H,(IY+5)
1360       PUSH HL
1370       LD   BC,(XPER)
1380       AND  A
1390       SBC  HL,BC
1400       LD   (IY+4),L
1410       LD   (IY+5),H
1420       LD   L,(IY+6)
1430       LD   H,(IY+7)
1440       PUSH HL
1450       LD   BC,(YPER)
1460       AND  A
1470       SBC  HL,BC
1480       LD   (IY+6),L
1490       LD   (IY+7),H
1500       CALL CREADA
1510       POP  HL
1520       LD   (IY+6),L
1530       LD   (IY+7),H
1540       POP  HL
1550       LD   (IY+4),L
1560       LD   (IY+5),H
1570       RET
1580 ;
1590 MAPA  LD   (NUBLBO),A
1600       LD   A,(MAIN)
1610       LD   E,A
1620       ADD  A,A
1630       ADD  A,A
1640       ADD  A,A
1650       LD   L,A
1660       LD   H,0
1670       LD   D,H
1680       LD   B,H
1690       ADD  HL,HL
1700       ADD  HL,DE
1710       LD   IY,TASPRI
1720       EX   DE,HL
1730       ADD  IY,DE
1740       LD   A,(SCROLL)
1750       CP   8
1760       JR   Z,NOAVA
1770       ADD  A,A
1780       LD   C,A
1790       LD   HL,DIRPE0
1800       ADD  HL,BC
1810       LD   E,(HL)
1820       INC  HL
1830       LD   D,(HL)
1840       LD   (IY+0),E
1850       LD   (IY+1),D
1860       LD   A,(VESCRO)
1870       AND  A
1880       JR   Z,NOAVA
1890       LD   HL,TAUME
1900       ADD  HL,BC
1910       LD   A,(HL)
1920       LD   C,A
1930       RLA
1940       SBC  A,A
1950       LD   B,A
1960       INC  HL
1970       LD   A,(HL)
1980       LD   E,A
1990       RLA
2000       SBC  A,A
2010       LD   D,A
2020       LD   L,(IY+4)
2030       LD   H,(IY+5)
2040       LD   A,(VESCRO)
2050 BUXM  ADD  HL,BC
2060       DEC  A
2070       JR   NZ,BUXM
2080       LD   C,L
2090       LD   B,H
2100       LD   L,(IY+6)
2110       LD   H,(IY+7)
2120       LD   A,(VESCRO)
2130 BUYM  ADD  HL,DE
2140       DEC  A
2150       JR   NZ,BUYM
2160       EX   DE,HL
2170       JR   PANDAV
2180 NOAVA LD   C,(IY+4)
2190       LD   B,(IY+5)
2200       LD   E,(IY+6)
2210       LD   D,(IY+7)
2220 PANDAV LD  A,B
2230       RLA
2240       JR   NC,NOREB
2250       LD   BC,0
2260 NOREB LD   A,D
2270       RLA
2280       JR   NC,NORED
2290       LD   DE,0
2300 NORED LD   HL,(MAPX)
2310       LD   H,0
2320       ADD  HL,HL
2330       ADD  HL,HL
2340       ADD  HL,HL
2350       AND  A
2360       SBC  HL,BC
2370       JR   NC,NOAB
2380       ADD  HL,BC
2390       LD   B,H
2400       LD   C,L
2410 NOAB  LD   HL,(MAPY)
2420       LD   H,0
2430       ADD  HL,HL
2440       ADD  HL,HL
2450       ADD  HL,HL
2460       AND  A
2470       SBC  HL,DE
```



```
2480        JR   NC,NOAD
2490        ADD  HL,DE
2500        EX   DE,HL
2510 NOAD   LD   (IY+4),C
2520        LD   (IY+5),B
2530        LD   (IY+6),E
2540        LD   (IY+7),D
2550        LD   A,(COLUPE)
2560        INC  A
2570        NEG
2580        LD   L,A
2590        LD   H,#FF
2600        INC  HL
2610        ADD  HL,BC
2620        LD   (XPER),HL
2630        LD   A,(FILAPE)
2640        INC  A
2650        NEG
2660        LD   L,A
2670        LD   H,#FF
2680        INC  HL
2690        ADD  HL,DE
2700        LD   (YPER),HL
2710        LD   HL,(NUMDEF)
2720        LD   A,H
2730        OR   L
2740        RET  Z
2750        LD   A,#C3
2760        LD   (CRECHA),A
2770        LD   DE,DIBMAP
2780        LD   (CRECHA+1),DE
2790        LD   IX,(TAMAPA)
2800 BUDECS PUSH HL
2810        LD   A,(IX+0)
2820        INC  A
2830        JR   Z,ESTENO
2840        LD   L,(IX+1)
2850        LD   H,0
2860        ADD  HL,HL
2870        ADD  HL,HL
2880        ADD  HL,HL
2890        LD   DE,(XPER)
2900        AND  A
2910        SBC  HL,DE
2920        LD   A,H
2930        AND  A
2940        JR   Z,PORDX
2950        INC  A
2960        JP   NZ,ESTENO
2970 PORDX  LD   (MTAB1+4),HL
2980        LD   L,(IX+2)
2990        LD   H,0
3000        ADD  HL,HL
3010        ADD  HL,HL
3020        ADD  HL,HL
3030        LD   DE,(YPER)
3040        AND  A
3050        SBC  HL,DE
3060        LD   A,H
3070        AND  A
3080        JR   Z,PORDY
3090        INC  A
3100        JP   NZ,ESTENO
3110 PORDY  LD   (MTAB1+6),HL
3120        LD   E,(IX+0)
3130        LD   D,0
3140        LD   IY,(TADECO)
3150        ADD  IY,DE
3160        EX   DE,HL
3170        ADD  HL,HL
3180        ADD  HL,HL
3190        EX   DE,HL
3200        ADD  IY,DE
3210        LD   A,(IY+0)
3220        LD   (MTAB2+2),A
3230        LD   A,(IY+1)
3240        LD   (MTAB2+3),A
3250        LD   A,(IY+2)
3260        LD   (MTAB1+2),A
3270        LD   A,(IY+3)
3280        LD   (MTAB1+3),A
3290        LD   A,(IY+4)
3300        LD   (AJUCOL+1),A
3310        PUSH IX
3320        LD   IX,MTAB2
3330        LD   IY,MTAB1
3340        CALL CREADA
3350        POP  IX
3360 ESTENO LD   BC,3
3370        ADD  IX,BC
3380        POP  HL
3390        DEC  HL
3400        LD   A,H
3410        OR   L
3420        JP   NZ,BUDECS
3430        LD   A,#E5
3440        LD   (CRECHA),A
3450        LD   HL,#7500
3460        LD   (CRECHA+1),HL
3470        RET
3480 ;
3490 DIBMAP LD   C,(IX+13)
3500        LD   A,(IX+14)
3510        LD   (BEAMOS+1),A
3520        NEG
3530        ADD  A,33
3540        LD   E,A
3550        LD   D,0
3560 AJUCOL LD   A,0
3570 BEAMOS LD   B,0
3580 BEBE   LD   (HL),A
3590        INC  HL
3600        DJNZ BEBE
3610        ADD  HL,DE
3620        DEC  C
3630        JR   NZ,BEAMOS
3640        LD   H,(IX+9)
3650        LD   L,#FF
3660        LD   A,(HL)
3670        LD   (RESOT+1),A
3680        INC  H
3690        LD   A,(HL)
3700        LD   (RESCA+1),A
3710        DEC  H
3720        LD   A,(IX+7)
3730        LD   (MNUMBY+1),A
3740        NEG
3750        ADD  A,33
3760        LD   (MSUMP+1),A
3770        LD   E,(IX+0)
3780        LD   D,(IX+1)
3790        EXX
3800        LD   L,(IX+2)
3810        LD   H,(IX+3)
3820        LD   E,(IX+8)
3830        LD   D,0
3840        LD   C,(IX+6)
3850 ZAMBL  EXX
3860        LD   A,(DE)
3870 RESCA  AND  0
3880        LD   C,A
3890        EXX
3900 MNUMBY LD   B,0
3910 MBBY   LD   A,(HL)
3920        EXX
3930        LD   L,A
3940        LD   A,(HL)
3950        OR   C
3960        LD   (DE),A
3970        INC  DE
3980        INC  H
3990        LD   C,(HL)
4000        DEC  H
4010        EXX
4020        INC  HL
4030        DJNZ MBBY
4040        EXX
4050        LD   A,(DE)
4060 RESOT  AND  0
4070        OR   C
4080        LD   (DE),A
4090        EX   DE,HL
4100 MSUMP  LD   C,0
4110        ADD  HL,BC
4120        EX   DE,HL
4130        EXX
4140        ADD  HL,DE
4150        DEC  C
4160        JR   NZ,ZAMBL
4170        RET
4180 ;
4190 TAUME  DEFB 1,0,0,1
4200        DEFB -1,0,0,-1
4210        DEFB 1,1,-1,1
4220        DEFB -1,-1,1,-1
4230 ;
4240 NCOMCH DI
4250        CALL COMCHO
4260        EI
4270        RET
4280 ;
4290 CHODEC DI
4300        LD   A,(23729)
4310        LD   L,A
4320        LD   H,0
4330        LD   B,H
4340        LD   C,L
4350        ADD  HL,HL
4360        ADD  HL,HL
4370        ADD  HL,HL
4380        ADD  HL,HL
4390        ADD  HL,BC
4400        LD   BC,TASPRI
4410        ADD  HL,BC
4420        PUSH HL
4430        POP  IY
4440        LD   HL,CHECK
4450        LD   A,(23728)
4460        LD   (CHEKIN+1),A
4470        INC  A
4480        JR   NZ,SICHEC
4490        LD   HL,DIRECC
4500 SICHEC LD   (SUBR+1),HL
4510        LD   IX,(TAMAPA)
4520        LD   BC,(NUMDEF)
4530 SUBR   CALL 0
4540        JR   Z,CHOCAR
4550        LD   DE,3
4560        ADD  IX,DE
4570        DEC  BC
4580        LD   A,B
4590        OR   C
4600        JR   NZ,SUBR
4610        LD   BC,65535
4620 CHOCAR EI
4630        RET
4640 ;
4650 CHECK  LD   A,(IX+0)
4660 CHEKIN CP   0
4670        JR   Z,DIRECC
4680        RET
4690 ;
4700 DIRECC LD   L,(IX+0)
4710        LD   H,0
4720        PUSH BC
4730        LD   B,H
4740        LD   C,L
4750        ADD  HL,HL
4760        ADD  HL,HL
4770        ADD  HL,BC
4780        LD   BC,(TADECO)
4790        ADD  HL,BC
4800        INC  HL
4810        INC  HL
4820        LD   B,H
4830        LD   C,L
4840        LD   L,(IX+1)
4850        LD   E,(IY+4)
4860        LD   D,(IY+5)
4870        XOR  A
4880        LD   (SALCHO+1),A
4890        LD   (SALCH2+1),A
4900        CALL UNADIM
4910        JR   NZ,OTRAV
4920        INC  BC
4930        LD   L,(IX+2)
4940        LD   E,(IY+6)
4950        LD   D,(IY+7)
4960        INC  IY
4970        LD   A,3
4980        LD   (SALCHO+1),A
4990        LD   (SALCH2+1),A
5000        CALL UNADIM
5010        DEC  IY
5020        JR   NZ,OTRAV
5030        POP  BC
5040        LD   HL,(NUMDEF)
5050        AND  A
5060        SBC  HL,BC
5070        LD   B,H
5080        LD   C,L
5090        XOR  A
5100        RET
5110 OTRAV  POP  BC
5120        XOR  A
5130        INC  A
5140        RET
5150 ;
5160 UNADIM LD   H,0
5170        ADD  HL,HL
5180        ADD  HL,HL
5190        ADD  HL,HL
5200        AND  A
5210        SBC  HL,DE
5220        JR   C,MAYSPR
5230        LD   A,H
5240        AND  A
5250        RET  NZ
5260        LD   A,(IY+2)
5270 SALCHO JR   SINADS
5280        ADD  A,A
5290        ADD  A,A
5300        ADD  A,A
5310 SINADS CP   L
5320        RET  C
5330        XOR  A
5340        RET
5350 MAYSPR INC  H
5360        RET  NZ
5370        LD   A,L
5380        NEG
5390        LD   L,A
5400        LD   A,(BC)
5410 SALCH2 JR   SINA2
5420        ADD  A,A
5430        ADD  A,A
5440        ADD  A,A
5450 SINA2  CP   L
5460        RET  C
5470        XOR  A
5480        RET
5490 ;
5500 MTAB1  DEFS 15
5510 MTAB2  DEFS 17
5520 ;
5530        ORG  52927
5540 ;
5550 TAMAPA DEFW 0
5560 NUMDEF DEFW 0
5570 TADECO DEFW 0
5580 NUMDEC DEFB 0
5590 TATABS DEFS 16
5600 TADEFS DEFS 16
5610 MAPACT DEFB 0
5620 BACK   DEFB 0
5630 MAIN   DEFB 0
5640 SCROLL DEFB 0
5650 MAPX   DEFB 0
5660 MAPY   DEFB 0
5670 COLUMN DEFB 0
5680 FILA   DEFB 0
5690 WIDSC  DEFB 0
5700 HIGHSC DEFB 0
5710 CALCUL DEFB 0
5720 FILAPE DEFB 0
5730 COLUPE DEFB 0
5740 DIRPE0 DEFW 0
5750 DIRPE1 DEFW 0
5760 DIRPE2 DEFW 0
5770 DIRPE3 DEFW 0
5780 DIRPE4 DEFW 0
5790 DIRPE5 DEFW 0
5800 DIRPE6 DEFW 0
5810 DIRPE7 DEFW 0
5820 VESCRO DEFB 4
5830 XPER   DEFW 0
5840 YPER   DEFW 0
5850 ;
5860 ;
5870 INSABO EQU  62318
5880 RELAP  EQU  62347
5890 VUELCA EQU  62383
5900 EXTIRC EQU  62389
5910 LOPPNU EQU  62404
5920 SITERC EQU  62488
5930 SALTO  EQU  62555
5940 CREADA EQU  62747
5950 CRECHA EQU  63135
5960 INICIO EQU  61953
5970 ORIGSC EQU  54568
5980 ORIGAT EQU  60904
5990 TASPRI EQU  53000
6000 LVOLCA EQU  62401
6010 COMCHO EQU  62082
6020 NUBLBO EQU  63389
```



LISTADO 5. ENSPRIT



```
10 *D+
20 *C-
30 ;
40 ;    SPRITES  POR
50 ;
60 ;   INTERRUPCIONES
70 ;
80 ;  POR PABLO ARIZA.
90 ;
100 ;
110         ORG  62194
120 ;
130 ;
140 ENTINT DI
150        PUSH AF
160        PUSH BC
170        PUSH DE
180        PUSH HL
190        PUSH IX
200        EXX
210        EX   AF,AF'
220        PUSH AF
230        PUSH BC
240        PUSH DE
250        PUSH HL
260        PUSH IY
270        LD   A,(ESTADO)
280        AND  A
290        JP   NZ,NOVOLC
300        CALL VUELCA
310        LD   A,1
320        LD   (ESTADO),A
330        JP   SALINT
340 ;
350 NOVOLC LD   HL,SPARES
360        XOR  A
370        LD   (ESTADO),A
380        LD   A,(NUBLBO)
390        LD   IX,TABLDI
400 INBOTO DEC  A
410        JR   Z,INSABO
420        EX   AF,AF'
430        LD   E,(IX+0)
440        LD   D,(IX+1)
450        LD   A,(IX+14)
460        LD   (BORPOI+1),A
470        LD   (BORPOD+1),A
480        NEG
490        ADD  A,33
500        LD   (AUMPOI+1),A
510        LD   (AUMPOD+1),A
520        LD   A,(IX+6)
530        LD   B,0
540 BORPOI LD   C,0
550        LDIR
560 AUMPOI LD   C,0
570        EX   DE,HL
580        ADD  HL,BC
590        EX   DE,HL
600        DEC  A
610        JP   NZ,BORPOI
620        LD   A,(IX+13)
630        LD   E,(IX+11)
640        LD   D,(IX+12)
650 BORPOD LD   C,0
660        LDIR
670 AUMPOD LD   C,0
680        EX   DE,HL
690        ADD  HL,BC
700        EX   DE,HL
710        DEC  A
720        JP   NZ,BORPOD
730        LD   C,15
740        ADD  IX,BC
750        EX   AF,AF'
760        JP   INBOTO
770 INSABO LD   A,1
780        LD   (NUBLBO),A
790        LD   DE,SPARES
800        LD   IX,TABLDI
810        LD   IY,TASPRI
820        LD   A,(NUMSPR)
830        LD   B,A
840        AND  A
850        JR   Z,NOSPRI
860 INMAIN PUSH BC
870        PUSH DE
880        CALL MOVERR
890        POP  DE
900 RELAP  CALL CREADA
910        LD   BC,17
920        ADD  IY,BC
930        POP  BC
940        DJNZ INMAIN
950        CALL DIBUJA
960 NOSPRI POP  IY
970        PUSH IY
980        RST  56
990 SALINT POP  IY
1000       POP  HL
1010       POP  DE
1020       POP  BC
1030       POP  AF
1040       EX   AF,AF'
1050       EXX
1060       POP  IX
1070       POP  HL
1080       POP  DE
1090       POP  BC
1100       POP  AF
1110       EI
1120       RETI
1130 ;
1140 ;
1150 VUELCA LD  HL,ORIGSC
1160       LD   DE,16384
1170 EXTIRC LD  C,0
1180       EXX
1190       LD   HL,ORIGAT
1200       LD   DE,22528
1210       LD   BC,25*256
1220 LVOLCA EXX
1230       LD   B,9
1240 LOPPNU LDI
1250       LDI
1260       LDI
1270       LDI
1280       LDI
1290       LDI
1300       LDI
1310       LDI
1320       LDI
1330       LDI
1340       LDI
1350       LDI
1360       LDI
1370       LDI
1380       LDI
1390       LDI
1400       LDI
1410       LDI
1420       LDI
1430       LDI
1440       LDI
1450       LDI
1460       LDI
1470       LDI
1480       LDI
1490       LDI
1500       LDI
1510       LDI
1520       LDI
1530       LDI
1540       LDI
1550       LDI
1560       LD   A,E
1570       SUB  32
1580       LD   E,A
1590       JR   C,NOINCD
1600       INC  D
1610 NOINCD INC HL
1620       DJNZ LOPPNU
1630       LD   A,E
1640       ADD  A,32
1650       LD   E,A
1660       JR   C,SITERC
1670       LD   A,D
1680       SUB  8
1690       LD   D,A
1700 SITERC EXX
1710       LDI
1720       LDI
1730       LDI
1740       LDI
1750       LDI
1760       LDI
1770       LDI
1780       LDI
1790       LDI
1800       LDI
1810       LDI
1820       LDI
1830       LDI
1840       LDI
1850       LDI
1860       LDI
1870       LDI
1880       LDI
1890       LDI
1900       LDI
1910       LDI
1920       LDI
1930       LDI
1940       LDI
1950       LDI
1960       LDI
1970       LDI
1980       LDI
1990       LDI
2000       LDI
2010       LDI
2020       LDI
2030       INC  HL
2040       DEC  B
2050 SALTO JP   NZ,LVOLCA
2060       RET
2070 ;
2080 ;
2090 MOVERR LD  A,(IY+5)
2100       CP   127
2110       RET  Z
2120       LD   A,(IY+9)
2130       LD   D,A
2140       AND  71
2150       LD   (IX+10),A
2160       BIT  4,D
2170       JR   Z,RECLIN
2180       LD   L,(IY+13)
2190       LD   H,(IY+14)
2200 RECUPE LD  A,(HL)
2210       CP   127
2220       JR   NZ,NOFITA
2230       LD   L,(IY+10)
2240       LD   H,(IY+11)
2250       LD   A,(HL)
2260 NOFITA CP  126
2270       JR   NZ,TRANOR
2280       INC  HL
2290       LD   A,(HL)
2300       LD   (IY+0),A
2310       INC  HL
2320       LD   A,(HL)
2330       LD   (IY+1),A
2340       INC  HL
2350       JP   RECUPE
2360 TRANOR LD  E,A
2370       INC  HL
2380       LD   A,(HL)
2390       INC  HL
2400       LD   (IY+13),L
2410       LD   (IY+14),H
2420       JP   COCORE
2430 RECLIN LD  A,(IY+10)
2440       LD   E,A
2450       LD   A,(IY+11)
2460 COCORE LD  C,A
2470       CP   128
2480       CCF
2490       SBC  A,A
2500       LD   B,A
2510       LD   L,(IY+6)
2520       LD   H,(IY+7)
2530       ADD  HL,BC
2540       LD   (IY+6),L
2550       LD   (IY+7),H
2560       LD   A,E
2570       LD   C,A
2580       CP   128
2590       CCF
2600       SBC  A,A
2610       LD   B,A
2620       LD   L,(IY+4)
2630       LD   H,(IY+5)
2640       ADD  HL,BC
2650       LD   (IY+4),L
2660       LD   (IY+5),H
2670       LD   A,(IY+12)
2680       LD   E,(IY+8)
2690       BIT  3,D
2700       JP   NZ,ADETRA
2710       INC  A
2720       CP   E
2730       JR   NZ,NOFICI
2740       XOR  A
2750 NOFICI LD  (IY+12),A
2760       JP   TRAREN
2770 ADETRA INC A
2780       LD   D,A
2790       SUB  E
2800       SUB  E
2810       ADD  A,2
2820       LD   (PARETR+1),A
2830       JR   Z,FIANCI
2840       LD   A,D
2850 FIANCI LD  (IY+12),A
2860       CP   E
2870       JR   C,TRAREN
2880 PARETR LD  A,0
2890       NEG
2900 TRAREN LD  E,(IY+15)
2910       LD   D,(IY+16)
2920       SLA  E
2930       RL   D
2940       LD   L,(IY+0)
2950       LD   H,(IY+1)
2960       AND  A
2970       JR   Z,PRIFAS
2980       LD   B,A
2990 MULTIP ADD HL,DE
3000       DJNZ MULTIP
3010 PRIFAS LD  (IX+2),L
3020       LD   (IX+3),H
3030       RET
3040 ;
3050 ;
3060 CREADA LD  A,(IY+4)
3070       LD   C,A
3080       AND  6
3090       ADD  A,248
3100       LD   (IX+9),A
3110       LD   A,(IY+5)
3120       AND  A
3130       JR   Z,XPOSIT
3140       INC  A
3150       RET  NZ
3160       LD   A,C
3170       ADD  A,6
3180       JR   NC,NEGARE
3190       LD   HL,ORIGAT-1
3200       LD   BC,ORIGSC-1
3210       LD   A,(IY+2)
3220       LD   (IX+7),A
3230       XOR  A
3240       LD   (IX+8),A
3250       JP   PARTEY
3260 NEGARE SCF
3270       RRA
3280       SRA  A
3290       SRA  A
3300       LD   B,A
3310       ADD  A,(IY+2)
3320       RET  NC
3330       RET  Z
3340       LD   (IX+7),A
3350       LD   A,B
3360       NEG
3370       LD   (IX+8),A
3380       ADD  A,(IX+2)
3390       LD   (IX+2),A
3400       LD   A,0
3410       ADC  A,(IX+3)
3420       LD   (IX+3),A
3430       LD   A,C
3440       LD   HL,ORIGAT-1
3450       LD   BC,ORIGSC-1
3460       AND  6
3470       JR   NZ,PARTEY
3480       INC  HL
3490       INC  BC
3500       JP   PARTEY
3510 XPOSIT LD  A,C
3520       SUB  1
3530       JR   C,INTEGE
3540       SRL  A
3550       SRL  A
3560       SRL  A
3570       ADD  A,(IY+2)
3580       CP   33
3590       JR   C,INTEGE
3600       SUB  32
3610       LD   (IX+8),A
3620       NEG
3630       ADD  A,(IY+2)
3640       LD   (IX+7),A
3650 RECOIN LD  A,C
3660       SRL  A
3670       SRL  A
3680       SRL  A
3690       LD   C,A
```

```
3700        LD   HL,ORIGAT
3710        ADD  A,L
3720        LD   L,A
3730        ADC  A,H
3740        SUB  L
3750        LD   H,A
3760        LD   A,C
3770        LD   BC,ORIGSC
3780        ADD  A,C
3790        LD   C,A
3800        ADC  A,B
3810        SUB  C
3820        LD   B,A
3830        JP   PARTEY
3840 INTEGE LD   A,(IY+2)
3850        LD   (IX+7),A
3860        XOR  A
3870        LD   (IX+8),A
3880        JP   RECOIN
3890 PARTEY LD   A,(IY+7)
3900        AND  A
3910        JR   Z,YPOSIT
3920        INC  A
3930        RET  NZ
3940        LD   A,(IY+6)
3950        ADD  A,(IY+3)
3960        RET  NC
3970        RET  Z
3980        LD   (IX+6),A
3990        NEG
4000        ADD  A,(IY+3)
4010        PUSH HL
4020        PUSH DE
4030        LD   L,(IX+2)
4040        LD   H,(IX+3)
4050        LD   D,0
4060        LD   E,(IY+2)
4070 NESUDI ADD  HL,DE
4080        DEC  A
4090        JP   NZ,NESUDI
4100        LD   (IX+2),L
4110        LD   (IX+3),H
4120        LD   A,(IX+6)
4130        ADD  A,7
4140        SRL  A
4150        SRL  A
4160        SRL  A
4170        LD   (IX+13),A
4180        JP   COYPOS
4190 YPOSIT LD   A,(IY+6)
4200        CP   192
4210        RET  NC
4220        SRL  A
4230        SRL  A
4240        SRL  A
4250        PUSH BC
4260        LD   BC,33
4270        INC  A
4280 CALATT DEC  A
4290        JR   Z,ATCOMP
4300        ADD  HL,BC
4310        JP   CALATT
4320 ATCOMP POP  BC
4330        PUSH HL
4340        PUSH DE
4350        LD   H,B
4360        LD   L,C
4370        LD   DE,33
4380        LD   A,(IY+6)
4390        AND  A
4400        JR   Z,PRIFIL
4410 CALFIL ADD  HL,DE
4420        DEC  A
4430        JP   NZ,CALFIL
4440        LD   B,H
4450        LD   C,L
4460 PRIFIL LD   A,(IY+6)
4470        ADD  A,(IY+3)
4480        JR   C,MOSCAB
4490        CP   193
4500        JR   NC,MOSCAB
4510        LD   A,(IY+3)
4520        LD   (IX+6),A
4530        LD   A,C
4540        EX   AF,AF'
4550        LD   C,(IY+6)
4560        SRL  C
4570        SRL  C
4580        SRL  C
4590        LD   A,(IY+6)
4600        ADD  A,(IY+3)
4610        ADD  A,7
4620        SRL  A
4630        SRL  A
4640        SRL  A
4650        SUB  C
4660        LD   (IX+13),A
4670        EX   AF,AF'
4680        LD   C,A
4690        JP   COYPOS
4700 MOSCAB LD   A,192
4710        SUB  (IY+6)
4720        LD   (IX+6),A
4730        ADD  A,7
4740        SRL  A
4750        SRL  A
4760        SRL  A
4770        LD   (IX+13),A
4780 COYPOS LD   (IX+0),C
4790        LD   (IX+1),B
4800        LD   L,(IX+2)
4810        LD   H,(IX+3)
4820        LD   E,(IY+15)
4830        LD   D,(IY+16)
4840        ADD  HL,DE
4850        LD   (IX+4),L
4860        LD   (IX+5),H
4870        LD   D,(IX+7)
4880        LD   A,(IX+9)
4890        AND  6
4900        JR   Z,SANUBY
4910        INC  D
4920 SANUBY LD   (IX+14),D
4930        POP  DE
4940        POP  HL
4950 CRECHA PUSH HL
4960        LD   (IX+11),L
4970        LD   (IX+12),H
4980        LD   H,B
4990        LD   L,C
5000        LD   A,(IX+14)
5010        LD   (PUBYSA+1),A
5020        LD   (PUATSA+1),A
5030        NEG
5040        ADD  A,33
5050        LD   (PUSUSA+1),A
5060        LD   (ATSUSA+1),A
5070        LD   A,(NUBLBO)
5080        INC  A
5090        LD   (NUBLBO),A
5100        LD   A,(IX+6)
5110        LD   B,0
5120 PUBYSA LD   C,0
5130        LDIR
5140 PUSUSA LD   C,0
5150        ADD  HL,BC
5160        DEC  A
5170        JR   NZ,PUBYSA
5180        POP  HL
5190        LD   A,(IX+13)
5200 PUATSA LD   C,0
5210        LDIR
5220 ATSUSA LD   C,0
5230        ADD  HL,BC
5240        DEC  A
5250        JP   NZ,PUATSA
5260        LD   C,15
5270        ADD  IX,BC
5280        RET
5290 ;
5300 ;
5310 DIBUJA LD   IX,TABLDI
5320        LD   A,(NUBLBO)
5330 DIBLUP DEC  A
5340        RET  Z
5350        PUSH AF
5360        LD   E,(IX+0)
5370        LD   D,(IX+1)
5380        LD   H,(IX+9)
5390        LD   L,255
5400        LD   A,(HL)
5410        LD   (PONUDO+1),A
5420        INC  H
5430        LD   A,(HL)
5440        DEC  H
5450        LD   (POSNUM+1),A
5460        EXX
5470        LD   L,(IX+2)
5480        LD   H,(IX+3)
5490        PUSH HL
5500        POP  IY
5510        LD   L,(IX+4)
5520        LD   H,(IX+5)
5530        LD   A,(IX+7)
5540        LD   (POSBYT+1),A
5550        NEG
5560        ADD  A,33
5570        LD   (SUMUAL+1),A
5580        LD   C,(IX+6)
5590        LD   E,(IX+8)
5600        LD   D,0
5610 LOSCAN EXX
5620 POSNUM LD   BC,0
5630        EXX
5640 POSBYT LD   B,0
5650 LOBYTS LD   A,(IY+0)
5660        INC  IY
5670        EX   AF,AF'
5680        LD   A,(HL)
5690        INC  HL
5700        EXX
5710        LD   L,A
5720        LD   A,(HL)
5730        OR   C
5740        INC  H
5750        LD   C,(HL)
5760        EX   AF,AF'
5770        LD   L,A
5780        LD   A,B
5790        LD   B,(HL)
5800        DEC  H
5810        OR   (HL)
5820        LD   L,A
5830        EX   AF,AF'
5840        EX   DE,HL
5850        AND  (HL)
5860        OR   E
5870        LD   (HL),A
5880        INC  HL
5890        EX   DE,HL
5900        EXX
5910        DJNZ LOBYTS
5920        ADD  HL,DE
5930        ADD  IY,DE
5940        EXX
5950        LD   A,C
5960 PONUDO OR   0
5970        EX   DE,HL
5980        AND  (HL)
5990        OR   B
6000        LD   (HL),A
6010 SUMUAL LD   A,0
6020        ADD  A,L
6030        LD   L,A
6040        ADC  A,H
6050        SUB  L
6060        LD   H,A
6070        EX   DE,HL
6080        EXX
6090        DEC  C
6100        JP   NZ,LOSCAN
6110        LD   L,(IX+11)
6120        LD   H,(IX+12)
6130        LD   A,(IX+14)
6140        LD   (ATDIRB+1),A
6150        NEG
6160        ADD  A,33
6170        LD   (SUMATT+1),A
6180        LD   E,(IX+10)
6190        LD   D,184
6200        LD   A,(IX+13)
6210 ATLAFX EX   AF,AF'
6220 ATDIRB LD   B,0
6230 ATBLDI LD   A,(HL)
6240        AND  D
6250        OR   E
6260        LD   (HL),A
6270        INC  HL
6280        DJNZ ATBLDI
6290 SUMATT LD   C,0
6300        ADD  HL,BC
6310        EX   AF,AF'
6320        DEC  A
6330        JP   NZ,ATLAFX
6340        POP  AF
6350        LD   C,15
6360        ADD  IX,BC
6370        JP   DIBLUP
6380 ;
6390 ;
6400 ESTADO DEFB 0
6410 NUBLBO DEFB 1
6420 ;
6430 ;
6440 DIMEXS LD   E,(IY+2)
6450        SLA  E
6460        SLA  E
6470        SLA  E
6480        JP   CONTIN
6490 DIMEYS LD   E,(IY+3)
6500 CONTIN LD   D,0
6510        AND  A
6520        SBC  HL,DE
6530        RET
6540 COORDS CALL DITABS
6550 CALPOI JP   0
6560 COORDX LD   L,(IY+4)
6570        LD   H,(IY+5)
6580        RET
6590 COORDY LD   L,(IY+6)
6600        LD   H,(IY+7)
6610        RET
6620 DITABS LD   IY,TASPRI
6630        AND  A
6640        RET  Z
6650        PUSH DE
6660        LD   DE,17
6670 SUMDIS ADD  IY,DE
6680        DEC  A
6690        JR   NZ,SUMDIS
6700        POP  DE
6710        RET
6720 ;
6730 ;
6740 NUMSPR EQU  23681
6750 TABLDI EQU  53289
6760 TASPRI EQU  53000
6770 ORIGSC EQU  54568
6780 ORIGAT EQU  60984
6790 SPARES EQU  53544
6800 SPRICH EQU  23728
6810 ;
6820 ;
6830        ORG  61953
6840 INICIO DI
6850        LD   A,241
6860        LD   I,A
6870        LD   B,0
6880        LD   HL,61696
6890 INTCRE LD   (HL),242
6900        INC  HL
6910        DJNZ INTCRE
6920        LD   (HL),242
6930        LD   DE,ORIGSC
6940        LD   HL,16384
6950        LD   A,192
6960 INILAP EX   AF,AF'
6970        PUSH HL
6980        LD   BC,32
6990        LDIR
7000        INC  DE
7010        POP  HL
7020        INC  H
7030        LD   A,H
7040        AND  7
7050        JR   NZ,INIPRT
7060        LD   A,L
7070        ADD  A,32
7080        LD   L,A
7090        JR   C,INIPRT
7100        LD   A,H
7110        SUB  8
7120        LD   H,A
7130 INIPRT EX   AF,AF'
7140        DEC  A
7150        JR   NZ,INILAP
7160        LD   HL,22528
7170        LD   DE,ORIGAT
7180        LD   A,24
7190 INATLA LD   BC,32
7200        LDIR
7210        INC  DE
7220        DEC  A
7230        JR   NZ,INATLA
7240        IM   2
7250        LD   C,4
7260        LD   H,248
7270 LCRTCR LD   A,9
7280        SUB  C
7290        SUB  C
7300        LD   L,0
7310 ROTACU LD   B,L
7320        LD   E,0
7330        LD   D,A
7340 VERROT DEC  D
7350        JR   Z,VEREXI
7360        SRL  B
7370        RR   E
7380        JR   VERROT
7390 VEREXI LD   (HL),B
7400        INC  H
7410        LD   (HL),E
7420        DEC  H
7430        DEC  L
7440        JR   NZ,ROTACU
7450        INC  H
7460        INC  H
7470        DEC  C
7480        JR   NZ,LCRTCR
7490        LD   A,1
7500        LD   (NUBLBO),A
7510        XOR  A
7520        LD   (NUMSPR),A
7530        LD   (ESTADO),A
7540        EI
7550        RET
7560 ;
7570 ;
7580 DESACT LD   A,63
7590        IM   1
7600        LD   I,A
7610        RET
7620 ;
7630 ;
7640 COMCHO LD   BC,(SPRICH)
7650        INC  C
7660        JR   Z,BCONTO
7670        DEC  C
7680        CALL CHOSUB
7690        LD   BC,0
7700        RET  NC
7710        INC  BC
7720        RET
7730 ;
7740 BCONTO LD   IX,SPRICH
7750        LD   A,(NUMSPR)
7760 BUCHAL INC  (IX+0)
7770        EX   AF,AF'
7780        LD   BC,(SPRICH)
7790        LD   A,B
7800        CP   C
7810        JR   Z,SASPRI
7820        CALL CHOSUB
7830        LD   BC,1
7840        RET  C
7850 SASPRI EX   AF,AF'
7860        DEC  A
7870        JR   NZ,BUCHAL
7880        DEC  BC
7890        RET
7900 ;
7910 CHOSUB LD   HL,COORDX
7920        LD   (CALPOI+1),HL
7930        LD   HL,DIMEXS
7940        LD   (ANCALT+1),HL
7950        CALL CHOREA
7960        RET  NC
7970        LD   HL,COORDY
7980        LD   (CALPOI+1),HL
7990        LD   HL,DIMEYS
8000        LD   (ANCALT+1),HL
8010 CHOREA LD   A,B
8020        CALL COORDS
8030        EX   DE,HL
8040        LD   A,C
8050        CALL COORDS
8060        AND  A
8070        SBC  HL,DE
8080        LD   A,B
8090        JR   NC,BIENCO
8100        LD   A,H
8110        CPL
8120        LD   H,A
8130        LD   A,L
8140        CPL
8150        LD   L,A
8160        INC  HL
8170        LD   A,C
8180 BIENCO CALL DITABS
8190 ANCALT JP   0
```

LISTADO 6. DEMOBASMAP

```
   1 BORDER 0: PAPER 0: INK 7: R
ANDOMIZE USR 51900: POKE 23681,1
  10 PAUSE 1: CLS : DIM C(4): FO
R X=1 TO 4: READ C(X): NEXT X: D
ATA 2,1,3,0
  15 POKE 23728,0: POKE 23729,0:
 LET A=USR 52690: IF A<20 THEN L
ET D=51695+3*A: IF PEEK 52969=3
THEN POKE D,6: PAUSE 1: PAUSE 1:
 PAUSE 1: PAUSE 1
  16 IF INKEY$=" " THEN RANDOMIZ
E USR 62075: STOP
  20 LET A$=INKEY$: IF A$>"8" OR
 A$<"5" THEN GO TO 15
  30 POKE 52969,C(VAL A$-4): GO
TO 15
9999 CLEAR 50958: LOAD "CODSPRIT
"CODE 61953,1495: LOAD "CODMAP"C
ODE 51900,989: LOAD "DEMOGRAF"CO
DE 50959,941: LOAD "DEMOVAR"CODE
 52927,90: RUN
```

LISTADO 8. DEMOVAR

Línea	Datos	Control
1	EFC9370094CA08EFC994	1441
2	CA94CA94CA94CA94CA94	1750
3	CA94CA37000000000000	607
4	00000000000000000000	0
5	500008403001011E1011	265
6	4078EFC76FC8EFC86FC9	1684
7	0FC70FC70FC70FC70238	914
8	018C00EFC70210B001FC	1074
9	00024600000100002000	105

DUMP: 52.927
N.° BYTES: 90

LISTADO 7. DEMOGRAF

Línea	Datos	Control
1	00000000010001000100	3
2	03800100038005400380	463
3	0D60007C0092007C01930	621
4	03800D60311803800380	575
5	03800380000000000000	262
6	0000000000300D881224	251
7	20940A400A4829042424	453
8	11880460000000000000	253
9	00000000000010100000	32
10	00000800000800800000	144
11	24000024000010400000	152
12	00000000000001001000	17
13	00000020220000000000	66
14	00440400100001100000	105
15	00000000000000000000	0
16	00000000000000000000	0
17	00002228292829481550	369
18	1550000000000000243C	197
19	3C3C180010103C080808	260
20	00000000000000000000	0
21	0000000000000006800B60	241
22	09D011A00BA033980380	899
23	000000000000001E1EFF	314
24	11702E802F605F9C40E2	987
25	7FFC00003BDC400216DA	964
26	40043BD80000FFE1E100	1048
27	C000C001800F80030001	660
28	00000001800380010000	261
29	0000000180003C0070000	331
30	001E1EFE11702E802F60	760
31	5F9C40E27FFC00003DEC	1217
32	400256DA40042F780000	605
33	FFE1C100C000C001800F	1201
34	80030001000000018001	262
35	80010000000000018003	261
36	C00700000002400DB01DB8	667
37	3E7C00002AA87FFE7FFE	1158
38	0FF020085554 7FFE7FFE	1226
39	2AA80000FDBFF00FE007	1140
40	C0038001C00380010000	648
41	00008001800100000000	258
42	00008001C00300000240	390
43	0DB01DB83E7C00002AA8	798
44	7FFE7FFE0FF0300C7FFE	1458
45	2AA855543FFC0000FDBF	1138
46	F00FE007C0038001C003	1005
47	80010000000080018001	387
48	000000000000008001C003	324
49	000078007F780E880174	634
50	06F439FA47023FFE0000	947
51	3BDC40025B68200021BDC	821
52	000087FF008700038003	659
53	F001C001800000008000	690
54	C0018001000000008000	450
55	C001E003000078007F78	787
56	0E88017406F439FA4702	897
57	3FFE000037BC40025B6A	823
58	20021EF4000087FF0087	833
59	00038003F001C0018000	696
60	00008000C00180010000	450
61	00008000C001E0030000	548
62	07E00FF010083FFC300C	885
63	6FF66FF6781E0FF02008	1159
64	55547FFE7FFE2AA80000	1141
65	F81FF00FE007C0038001	1089
66	80010000000000008001	258
67	80010000000000008001	258
68	C00300007E00FF01008	705
69	3FFC300C6FF66FF6781E	1239
70	0FF0300C7FFE2AAA5554	1077
71	3FFC0000F81FF00FE007	1080
72	C0038001800100000000	453
73	00008001800100000000	258
74	00008001C00300000000	324
75	000A00022A000A16000C	98
76	0400120C001222001C02	116
77	00201A00220E00242200	176
78	2804002A120032000032	204
79	2C0034080003810003C02	238
80	003A1C003E2C04020604	208
81	0A2A04102E0412040412	166
82	12042004042E08043E0C	194
83	043E1E01060401021001	127
84	08240116080011A1C012C	175
85	2201380202021803080C	144
86	02101A03162C021C1003	162
87	1C24022A1A022C2C0330	275
88	14023802033820050C0C	200
89	07181605261807222E05	212
90	262807321A0538260073C	327
91	0A0FC70218543FC70210	614
92	555FC70210557FC70210	826
93	569FC70210548FC70110	953
94	55CFC70210548FC70110	1000
95	56000000000000000000	86

DUMP: 50.959
N.° BYTES: 941

La Demo puede ser utilizada para ver otros mapas, tan sólo quitando la línea 9999 y cargando los datos del mapa que hemos creado.





''En la demo podremos observar la utilidad que pueden tener las rutinas de detección de choques''



Gráficos utilizados



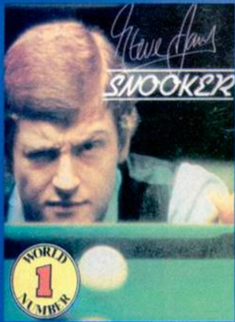
EL JOYSTICK MAS PREMIADO EN EUROPA
— más de 100.000 unidades vendidas en 6 meses
EL UNICO JOYSTICK QUE SE ADAPTA PERFECTAMENTE A LA MANO DEL JUGADOR.
• EL KONIX SPEEDKING UTILIZA EL MAS AVANZADO MICROSWITCH DE ORIGEN SUIZO
CAPAZ DE SOPORT
MAS DE 10.000.0
MOVIMIENT
• GARANTIA
6 MES
Si no lo encuentras en tu establecimiento habitual, pídelo a SERMA: Cardenal Belluga, 21. 28028 Madrid. Tels. 256 21 01 - 02
SER



5 Especiales por 990 ptas.
HAZTE UN REGALO ESPECIAL
COMPLETA TU COLECCION
Agotado el Especial nº 1
Recorta o copia este cupón y envíalo a Hobby Press, S.A. Apartado de Correos nº 232. 28100 Alcobendas (Madrid)
Deseo recibir en mi domicilio los ESPECIALES números 2, 3, 4, 5, y 6 de MICROHOBBY al precio de 990 ptas.
Deseo recibir en mi domicilio los ESPECIALES MICROHOBBY números ... al precio de 250 ptas. cada uno
Nombre
Apellidos ... Fecha de nacimiento
Domicilio
Localidad ... Provincia
C. Postal ... Teléfono ... (Para agilizar tu envío, es importante que indiques el código postal)
Formas de Pago
Talón bancario adjunto a nombre de Hobby Press, S. A.
Giro Postal a nombre de Hobby Press, S. A. nº
Fecha de caducidad de la tarjeta
Nombre del titular (si es distinto)
(Si pago con tarjeta de crédito, recibiré un nº más de regalo)
Fecha y firma







Trivial Pursuit™

MUCHO MAS QUE UN JUEGO

Disponible en

SPECTRUM
COMMODORE
AMSTRAD
AMSTRAD - DISC

Y ahora, también disponible en

PLUS + 3

ZAFIRO SOFTWARE DIVISION. SILVA 6, 28013 MADRID.
Tel 2419424 - 2419625 Telex 22690 ZAFIR E.

CARGADOR UNIVERSAL DE CODIGO MAQUINA

Utilización. En la línea inferior de la pantalla, aparecerá un pequeño menú de opciones a cada una de las cuales se accede pulsando la tecla que corresponde con su inicial:

INPUT. Este comando sirve para introducir nuevas líneas de Código Fuente. Al pulsarlos, el programa nos solicita un número de línea. Obligatoriamente, hemos de comenzar por la línea 1 a no ser que ya hayamos introducido alguna otra previamente.

Tras indicar el número de línea, nos pedirá los datos correspondientes a la misma. Una vez tecleados, y suponiendo que no haya habido ningún error hasta el momento, hay que introducir el Control, que está situado en cada línea, pudiendo pasar, si lo deseamos, al menú principal pulsando simplemente **«ENTER»**.

TEST. Para listar por pantalla las líneas de datos que hayamos metido hasta el momento.

DUMP. Este comando vuelca el contenido de la variable A$ en memoria, a partir de la dirección que se especifique. Esta operación es obligatoria antes de hacer funcionar una rutina o programa en código máquina. En la mayoría de los casos, con la rutina se indicará también la dirección de memoria donde debe ser volcada y su longitud expresada en bytes.

Al intentar volcar el código fuente, puede ocurrir que nos aparezca el mensaje **«ESPACIO DE TRABAJO»**. Esto indica que estamos intentando volcar en una zona que el ordenador está usando para sus propios cálculos.

SAVE. Este comando nos permite salvar en cinta el código fuente o el código objeto para su posterior utilización. Al pulsar SAVE nos aparecerá un segundo menú de tres opciones: Salvar Código Fuente (F), Salvar Código Objeto (O), indicando dirección y número de bytes, o volver al menú principal (R).

LOAD. Cuando el número de datos a teclear sea grande, es normal tener que realizar el trabajo en varias veces. Para ello, puede salvarse en cinta la parte que tengamos *(Código Objeto)* y luego recuperar mediante la opción LOAD.

Una vez tecleado el programa cargador hay que hacer GOTO 99ØØ, con lo que se grabará y verificará en cinta.

Si por cualquier razón, intencionada o no, se detuviese durante su utilización, es imprescindible teclear «GOTO menu», nunca RUN ni ningún tipo de CLEAR, ya que estos dos comandos destruyen las variables y con ellas el código fuente que hubiera almacenado hasta el momento.

Las líneas que no aparezcan deben teclearse con 2Ø ceros como dato y Ø como control.

```
   2 REM [illegible]
   3 REM CARGADOR CM MICROHOBBY
   4 REM [illegible]
   5 CLEAR 65535: LET menu=6000
  10 FOR n=23296 TO 23312
  12 READ c: POKE n,c: NEXT n
  15 DATA 42,75,92,126,254,193,40,6,205,104,25,235,24,245,54,65,201
  70 LET a$="": POKE 23658,8
 100 LET a=10: LET b=11: LET c=12: LET d=13: LET e=14: LET f=15
 200 LET li=1: GO TO 6000
1000 REM [illegible]
1001 INPUT "LINEA", LINE l$: IF l$="" THEN GO TO 6000
1002 FOR n=1 TO LEN l$
1003 IF l$(n)<"0" OR l$(n)>"9" THEN GO TO 1001
1004 NEXT n: LET line=VAL l$
1005 IF line<>li THEN POKE 23689,PEEK 23689-1: GO SUB 5000: GO TO 1000
1007 INPUT "[illegible] DATOS [illegible]"; LINE d$
1008 IF d$="" THEN GO TO 6000
1009 LET cx=24-PEEK 23689: PRINT AT cx,0;d$;AT cx,21;CHR$ 138;"LINEA ";li
1010 IF LEN d$<>20 THEN GO SUB 5000: GO TO 1000
1020 FOR n=1 TO 20
1110 LET w$=d$(n)
1150 IF w$>CHR$ 47 AND w$<CHR$ 58 OR w$>CHR$ 64 AND w$<CHR$ 71 THEN GO TO 1170
1160 PRINT AT cx,n-1; FLASH 1; OVER 1;" ": GO SUB 5000: GO TO 1000
1200 NEXT n: LET ch=0
1210 FOR n=1 TO 20 STEP 2
1215 LET he=VAL d$(n)*16+VAL d$(n+1): LET ch=ch+he: NEXT n
1250 LET ct=0: INPUT "CONTROL ";ct
1260 IF ct<>ch THEN GO SUB 5000: GO TO 1000
1300 LET a$=a$+d$
2000 LET li=li+1: GO TO 1000
5000 BEEP .2,0: OUT 254,2: POKE 23689,PEEK 23689+1: RETURN
6000 REM [illegible]
6005 PRINT #0; INK 7; PAPER 1;" INPUT LOAD SAVE DUMP TEST "
6100 LET i$=INKEY$: IF i$="" THEN GO TO 6100
6200 IF i$="I" THEN GO TO 1000
6210 IF i$="S" THEN GO TO 7000
6220 IF i$="L" THEN GO TO 8000
6225 IF i$="T" THEN GO TO 7500
6230 IF i$="D" THEN GO TO 9000
6250 GO TO 6100
7000 REM [illegible]
7001 PRINT #0; PAPER 3; INK 7;" FUENTE(F) OBJETO(O) RETURN(R) "
7002 PAUSE 0: IF INKEY$<>"F" AND INKEY$<>"O" AND INKEY$<>"R" THEN GO TO 7002
7003 IF INKEY$="O" THEN GO TO 7250
7004 IF INKEY$="R" THEN CLS : GO TO 6000
7005 REM [illegible]
7006 IF a$="" THEN GO SUB 9500: GO TO 6000
7008 RANDOMIZE li
7010 LET a$=CHR$ PEEK 23670+CHR$ PEEK 23671+a$
7015 INPUT "NOMBRE (Save)", LINE n$: IF n$="" OR LEN n$>10 THEN GO TO 7015
7020 SAVE n$ DATA a$()
7025 PRINT #0; PAPER 6;" DESEA VERIFICAR (S/N) ": PAUSE 0: IF INKEY$="S" THEN PRINT #0; INK 7; PAPER 2;" REBOBINE LA CINTA Y PULSE PLAY ": VERIFY n$ DATA a$(): CLS : PRINT "CODIGO FUENTE: ";n$: PAUSE 200
7030 LET a$=a$(3 TO ): CLS
7040 GO TO 6000
7250 REM [illegible]
7255 INPUT PAPER 3; INK 7;"DIRECCION ";di; PAPER 3; INK 7;"N.BYTES ";nb
7260 INPUT "NOMBRE (Save)", LINE n$: IF n$="" OR LEN n$>10 THEN GO TO 7260
7270 SAVE n$CODE di,nb
7275 PRINT #0; PAPER 6;" DESEA VERIFICAR (S/N) ": PAUSE 0: IF INKEY$="S" THEN PRINT #0; INK 7; PAPER 2;" REBOBINE LA CINTA Y PULSE PLAY ": VERIFY n$CODE di,nb: CLS : PRINT "CODIGO OBJETO: ";n$'"Inicio: ";di,"Longitud: ";nb: PAUSE 200
7280 CLS
7300 GO TO 6000
7500 REM [illegible]
7503 IF a$="" THEN GO SUB 9500: GO TO 6000
7505 CLS : FOR m=1 TO (LEN a$) STEP 20
7510 PRINT a$(m TO m+19);" ";CHR$ 138;"LINEA ";INT (m/20)+1
7520 NEXT m: GO TO 6000
8000 REM [illegible]
8010 INPUT "NOMBRE (Load)", LINE n$
8020 LOAD n$ DATA a$()
8025 RANDOMIZE USR 23296
8030 LET li=CODE a$(1)+256*CODE a$(2): LET a$=a$(3 TO )
8035 CLS : PRINT AT 10,5;"Ultima linea: ";li-1;AT 11,5;"Comenzar por: ";li
8040 GO TO 6000
9000 REM [illegible]
9003 IF a$="" THEN GO SUB 9500: GO TO 6000
9005 INPUT "DIRECCION ";di: CLS
9006 IF di<PEEK 23653+256*PEEK 23654 OR (di+LEN a$/2)>65300 THEN PRINT FLASH 1;AT 5,6;"ESPACIO DE TRABAJO": FOR n=1 TO 200: NEXT n: CLS : GO TO 6000
9007 PRINT AT 9,7; FLASH 1;"VOLCANDO EN MEMORIA": PRINT AT 7,5;"Direccion Inicial: ";di
9008 PRINT AT 11,4;"[illegible]";AT 11,17;"[illegible]"
9010 FOR n=1 TO (LEN a$) STEP 2
9015 POKE di,VAL a$(n)*16+VAL a$(n+1): LET di=di+1
9018 PRINT AT 11,12;INT (LEN a$/2-n/2);" "
9020 NEXT n: CLS : PRINT AT 10,8; FLASH 1;"[illegible]": FOR n=1 TO 100: NEXT n: CLS : GO TO 6000
9500 REM [illegible]
9501 CLS : PRINT #0; FLASH 1;" NO EXISTE NINGUN CODIGO FUENTE ": PAUSE 300: CLS : RETURN
9900 CLEAR : SAVE "CARGADOR" LINE 1: PRINT #0;"REBOBINE LA CINTA PARA VERIFICAR": VERIFY "CARGADOR": RUN
```

Cómo utilizar el Cargador Universal

La mayoría de los Patas Arriba destinados a Spectrum constan de dos listados que se complementan entre sí. Para utilizar estos programas debemos seguir los siguientes pasos:

1. El programa Cargador Universal debe estar grabado en una cinta. Si no es así lo teclearemos y lo guardaremos en una cinta, usando SAVE «CARGADOR» LINE Ø.

2. Observamos los listados, ya que éstos se numeran según el orden en el que deben ser utilizados.

3. Tecleamos el listado 1 o el programa en Basic y lo salvamos en una cinta virgen utilizando: SAVE «(NOMBRE)» LINE 1. Donde nombre será cualquier dato que nos ayude a encontrarlo en la cinta y utilizaremos LINE para que el programa se autoejecute automáticamente al terminar la carga. Esta fórmula también puede ser utilizada cuando el cargador tenga un solo bloque.

4. Cargamos el programa Cargador Universal en el ordenador, y cuando aparezca el menú de opciones (INPUT, LOAD, SAVE, DUMP, TEST) elegiremos la opción INPUT pulsando la tecla I.

5. A partir de este momento procederemos a introducir el listado 2 o de datos. En la parte inferior de la pantalla aparece la palabra línea, entonces teclearemos la línea 1 y pulsaremos ENTER.

6. Entonces introduciremos los datos, que serán siempre los 20 números y letras que se encuentran a continuación. Al terminar pulsaremos ENTER.

7. El ordenador nos pedirá el control, que es la suma de todos los datos, y daremos el número que se encuentra al final de la línea. En este proceso debemos tener mucho cuidado, ya que si nos equivocamos tendremos que introducir de nuevo la línea.

8. Repetiremos esta operación hasta que tecleemos todas las líneas.

9. Cuando hayamos tecleado todas las líneas, nos seguirá pidiendo otra línea, pulsaremos ENTER directamente y aparecerá de nuevo el menú.

10. Procederemos a realizar el DUMP, que no es otra cosa que un volcado en la memoria de los datos anteriormente introducidos.

11. El DUMP figura siempre cerca del listado en formato Cargador Universal, pero si éste no apareciera, o por cualquier problema el programa no lo permitiera, es aconsejable indicar la dirección 4ØØØØ. Para elegir esta opción basta con pulsar la tecla D.

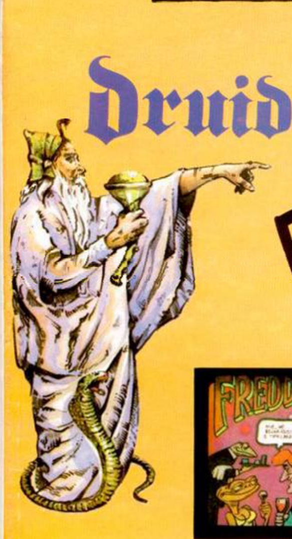
12. Una vez realizado volverá a aparecer el menú de opciones. Eligiendo la opción SAVE nos aparecerá un nuevo menú (FUENTE, OBJETO, RETURN); si el listado es muy extenso y no lo hemos terminado podemos continuar en otro momento donde lo dejamos usando el Fuente, pero si hemos terminado y realizado el DUMP, salvaremos el código objeto pulsando la tecla O.

13. El programa nos volverá a pedir la dirección. En ella daremos la misma que utilizamos para el DUMP. Luego nos pedirá el número de bytes que deseamos salvar a partir de la dirección indicada; este dato también figura al lado del DUMP. Si no aparece debemos contar el número de líneas y multiplicarlas por diez. Por último, el programa nos pedirá el nombre con el que deseamos grabar el bloque de bytes.

14. En la pantalla aparecerá Start tape, then press any key. Pondremos la cinta donde grabamos el listado 1 y pulsando una tecla grabaremos el listado 2.

15. Ya sólo nos hace falta cargar lo que hemos grabado en el cassette, y contestar a las preguntas.

Empieza la Aventura con el



EXTRA MICRO Manía

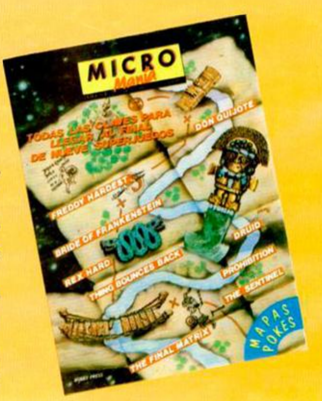
EXTRA N3 Sólo para adictos 400 Ptas.











Todo un extra donde ponemos "PATAS ARRIBA" los juegos más populares del momento. Freddy Hardest, Don Quijote, The Sentinel y muchos más; con más mapas y pokes que nunca, para los adictos como tú.
Si has empezado la aventura, con el EXTRA MICROMANÍA, llega al final sin perder pantalla.

¡A LA VENTA EN TU KIOSKO!

¡NO!

NO HAY NADA MEJOR

SPECTRUM +3 PACK

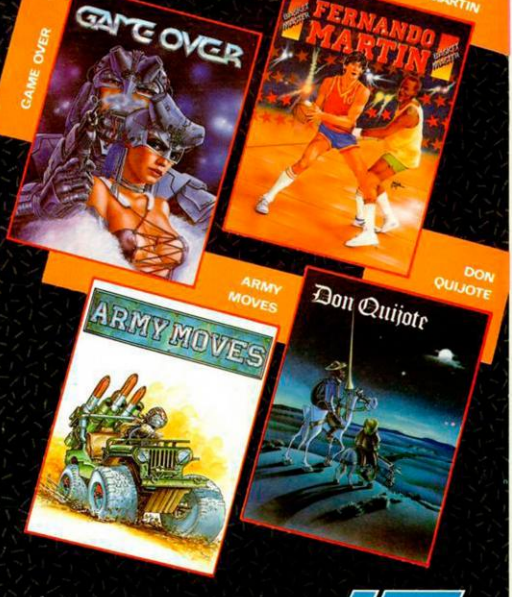
disco

GAME OVER



F. MARTIN





No hay nada mejor, aquí tienes las pruebas:

NO hay ningún programa más vendido en la historia de España que ***FERNANDO MARTIN BASKET MASTER.***

NO existe otro juego que haya dado más que hablar en Europa que ***GAME OVER.***

NO se ha superado el record de permanencia en el TOP 20, establecido por ***ARMY MOVES.***

NO podía imaginarse nunca el éxito de una aventura gráfico-conversacional como ***DON QUIJOTE.***

NO pierdas la oportunidad.

Ahora tienes los cuatro ***MEGA-JUEGOS*** en un disco.



ARMY MOVES



DON QUIJOTE

DINAMIC SOFTWARE. PZA. DE ESPAÑA, 18. TORRE DE MADRID, 29-1. 28008 MADRID.
TELEX: 44124 DSOFT-E TIENDAS Y DISTRIBUIDORES: (91) 314 · 18 · 04.
PEDIDOS CONTRA REEMBOLSO: (91) 248 · 78 · 87.

DINAMIC